2017

Contributions to improve the accuracy and computational efficiency of genomic prediction

Hao Cheng
Iowa State University

Contributions to improve the accuracy and computational efficiency of genomic prediction

by

Hao Cheng

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Co-majors: Genetics; Statistics

Program of Study Committee:
Rohan L. Fernando, Co-major Professor
Dorian J. Garrick, Co-major Professor
Alicia L. Carriquiry, Co-major Professor
Jack C.M. Dekkers
Jarad B. Niemi

Iowa State University

Ames, Iowa

2017

To the Vet, the Cat and the Dog

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I cannot imagine how lucky I was to happen to walk into Dr. Rohan Fernandos office during my first-year research rotation, and choosing Rohan as my major professor probably is the best choice I made during my PhD studies. I am deeply indebted to my another excellent major professor, Dr. Dorian Garrick, for his extremely helpful guidance and inspiration. I benefit undoubtedly, greatly and enormously from working closely with Rohan and Dorian.

I also would like to express my gratitude to my co-major professor in statistics, Dr Alicia Carriquiry, from whom I took my first serious statistical class. Dr. Carriquiry made my desire to study statistics as a co-major come true, which gives me strength and confidence in my statistical research.

I am also deeply thankful to Dr. Jack Dekkers. The very first quantitative genetics class that I took was his ANS561, and I finally got an A in that class. I guess he has no idea how important that A was for me, a new graduate student who was building his confidence in research. I am also full of thankfulness to Dr. Jarad Niemi. I really enjoyed those case studies he had in his class for Bayesian methods.

Last but not least, I would like to express my thanks to my wife, Shen Li. I wouldnt have achieved what I did without her. I also want to thank my pets, Ottaman, the naughty Labrador, and Oktty, the grumpy cat. Thank Oktty for walking on my keyboard but carefully avoiding the delete key. Thank Ottaman for only ingesting toxic stuff while I was not in charge of him (I was not in charge most of the time anyway, plus my wife is a veterinarian).

ABSTRACT

The discovery of genome-wide high-density molecular markers (e.g., single-nucleotide polymorphisms, SNPs) has revolutionized genetic analyses in human medicine, animal and plant breeding. There are several active areas of research and development in whole-genome analyses, including 1) collection or simulation of genomic data, 2) use of genomic data for prediction or genome-wide association studies, and 3) validation of the performance of these analyses. In this thesis, several statistical models and computational algorithms were proposed and investigated, contributing to these three areas of research and development.

A contribution to the first area is a simulation strategy that drops down origins and positions of chromosomal segments rather than every allele state to efficiently simulate sequence data and complex pedigree structures across multiple generations. A software tool called XSim, which incorporates the efficient strategy, was developed with implementations in C++ and Julia. XSim allows the genome of founders to be characterized by real genome sequence data and complex pedigree structures among descendants.

Several methods contributing to the use of genomic data for prediction and genome-wide association studies (GWAS) were proposed and investigated. Two methods were proposed to improve the computational efficiency of Bayesian multiple-regression analyses. First, we showed how Gibbs samplers without the use of the Metropolis-Hastings (MH) algorithm can be used for the BayesB method, where the prior for each marker effect follows a mixture distribution with a point mass at zero with probability π and a univariate-t distribution with probability $1-\pi$. We showed that by introducing a variable δ_j in BayesB, indicating whether the marker effect for a locus is zero or non-zero, the marker effect and locus-specific variance can be sampled using Gibbs. We considered three different versions of the Gibbs sampler to sample each marker effect, locus-specific variance and its indicator variable δ_j. Computational efficiencies defined as the number of effective samples per second of computing time were compared with simulated

data. Among the Gibbs samplers that were considered, the most efficient sampler is about 2.1 times as efficient as the MH algorithm proposed by Meuwissen et al. and 1.7 times as efficient as that proposed by Habier et al. Second, we proposed a strategy to parallelize Gibbs sampling for each marker within each step of the MCMC chain. This parallelization is accomplished by using an orthogonal data augmentation strategy, where the marker covariate matrix is augmented by adding p new rows, where p is the number of markers, such that its columns are orthogonal. The use of this strategy is expected to increase the speed of Gibbs sampling with lower memory requirements. The parallel Gibbs sampling approach using an augmented marker covariate matrix was shown for BayesC methods, where the prior for each marker effect follows a mixture distribution with a point mass at zero and a univariate normal distribution. The full conditional distributions that are needed for BayesC with orthogonal data augmentation (BayesC-ODA) were derived and the convergence of BayesC-ODA was studied. In analyses of the simulated data, BayesC-ODA provided virtually identical predictions of breeding values as BayesC when the chain length was about 20,000 to 80,000, which is similar to the commonly used chain length of 50,000.

Two methods were proposed or investigated to improve prediction accuracy of Bayesian multiple- regression analyses. First, we proposed a flexible variable selection model for multiple-trait analyses with BayesCπ or BayesB priors. This model was compared to single-trait methods and a previously proposed multi-trait model using real and simulated data. Flexible variable selection showed an advantage when data were from two simulated traits, where a locus had an effect only on one of the traits. Second, we compared alternative approaches to single-trait genomic prediction using genotyped and non-genotyped Hanwoo beef cattle. In those data analyses, the single-step methods, which take advantage of all pedigree, phenotypic and genomic information simultaneously, gave similar or higher prediction accuracies compared to methods using only genotyped or non-genotyped individuals. Alternative priors allowed single-step Bayesian regression methods (SSBR) to outperform single-step genomic best linear unbiased prediction (SSGBLUP) in some cases.

One method contributing to the validation of the performance of whole-genome analyses was proposed. In leave-one-out cross validation (LOOCV), one individual is omitted for training

with validation on the omitted individual. Efficient LOOCV strategies were proposed for genomic best linear unbiased prediction (GBLUP) in scenarios when $n > p$ or $n < p$, where n is the number of observations and p is the number of markers. These strategies were compared to naive application of LOOCV with simulated data. In these data analyses, efficient LOOCV, requiring little more effort than a single analysis, was much faster than the naive LOOCV.

CHAPTER 1. GENERAL INTRODUCTION

1.1 Introduction

The discovery of genome-wide high-density molecular markers (e.g., single-nucleotide polymorphisms, SNPs) has revolutionized genetic analyses of quantitative traits in human medicine (de los Campos et al., 2010; Makowsky et al., 2011; Vazquez et al., 2012; Spiliopoulou et al., 2015), animal (VanRaden, 2008; Hayes et al., 2009; VanRaden et al., 2009; Habier et al., 2010b; Wolc et al., 2012) and plant breeding (Crossa et al., 2010). Genomic prediction was proposed by Meuwissen et al (Meuwissen et al., 2001a) to incorporate marker effects from whole-genome data with phenotypic data into genetic evaluation in animal and plant breeding, and this is the primary application that uses whole-genome data in agriculture. In genomic prediction, all the marker effects are estimated simultaneously, and estimates of marker effects are then used to predict the breeding values, which is defined as the sum of the effects of all the markers, of selection candidates. Another use of whole-genome data is genome-wide association studies (GWAS). In GWAS, the association between molecular markers and phenotypes is assessed, where a single marker (Maher, 2008; Manolio et al., 2009; Visscher et al., 2010) or genomic window (Sahana et al., 2010; Hayes et al., 2010; Fernando et al., 2017) is tested at a time or simultaneously. There are several active areas of research and development in whole-genome analyses, including collection or simulation of genomic data, use of genomic data for prediction or GWAS, and validation of the performance of these analyses.

1.1.1 Data collection or simulation

One of the first steps in whole-genome analyses is collection or simulation of data. Due to advances in high-throughput genotyping and sequencing technologies, real or imputed high-

density SNP genotypes are routinely used for genomic prediction and genome-wide association studies, and many researchers are moving towards the use of actual or imputed next generation sequence data in whole-genome analyses. Analysis of real or imputed genotypes for genomic prediction and genome-wide association studies, however, can result in findings that are difficult to validate. On the other hand, simulated data have advantages in that the underlying causal mutations and simulated breeding values are available for direct validation. In general, there are two types of simulation methods: coalescent methods and forward-in-time (drop down) methods. Compared to coalescent-based simulations, forward-in-time simulations are computationally intensive but very flexible, which allows modeling large numbers of recombination events in concert with complex life-like selection scenarios (Chadeau-Hyam et al., 2008; Hoggart et al., 2007).

1.1.2 Use of genomic data for prediction and GWAS

An important aspect of whole-genome analyses is prediction of unobserved genotypic or breeding values using information from phenotypes, genotypes or pedigree. Before high-density marker panels were available, only pedigree information was used for prediction in animal and plant breeding (Henderson, 1984). A widely used statistical method to incorporate pedigree information into genetic evaluation is best linear unbiased prediction (BLUP). In pedigree-based BLUP, unobserved breeding values are included in a mixed linear model as random effects, where the covariance between breeding values of relatives is proportional to the identical by decent probability (IBD) between the relatives, which is the probability that alleles drawn at random from the same locus of the two relatives originated from the same allele of a common ancestor. BLUP can be efficiently obtained by solving Henderson's mixed model equations (MME) corresponding to this mixed linear model (Henderson, 1984).

Since the availability of genome-wide SNP panels, genomic prediction has been adopted for improvement of livestock and is rapidly replacing pedigree-based BLUP. Following the principle that is used in pedigree-based BLUP, a widely-used statistical method to incorporate genotypic information is genomic BLUP, where unobserved breeding values are fitted as random effects based on covariances defined by a genomic relationship matrix computed from

genotypes (Nejati-Javaremi et al., 1997) instead of pedigree. An alternative but equivalent model (Fernando, 1998; Strandén and Garrick, 2009) for genomic BLUP is the use of a random multiple-regression model that simultaneously fits marker effects as uncorrelated random effects, known as random regression BLUP (Meuwissen et al., 2001a). Random regression BLUP is computationally more efficient when the number n of individuals is larger than the number p of markers in the model, because for this model the MME are of order about p.

Random regression BLUP can be viewed as a special case of whole-genome Bayesian multiple-regression methods (Meuwissen et al., 2001a), which are widely used to address the problem that the number p of marker covariates is usually larger than the number n of observations. In Bayesian multiple regression methods, the effects of all markers are estimated simultaneously combining information from the phenotypic data and priors for the marker effects. Bayesian multiple-regression methods were first proposed for genomic prediction (Meuwissen et al., 2001a), but can also be adapted for GWAS (Fernando et al., 2017). The primary difference between these methods is the prior assumed for the effects of the covariates (Gianola, 2013). For example, the prior for each marker effect in BayesA (Meuwissen et al., 2001a) follows a scaled t distribution, and BayesA can be viewed a special case of BayesB, where the prior for each marker effect follows a mixture distribution with a point mass at zero with probability π and a univariate-t distribution with probability $1 - \pi$ (Gianola, 2013). When $\pi = 0$, BayesB becomes BayesA. Another widely-used Bayesian mixture model is BayesC, in which a common variance is used for all SNPs instead of locus-specific variances (Kizilkaya et al., 2010), and a modification of that method known as BayesCπ treats π as an unknown parameter with a uniform prior distribution (Habier et al., 2011b).

Incorporating prior information to whole-genome analyses helps to improve the prediction accuracy. Another approach to increase accuracy is to borrow information from other sources of data such as the use of multiple trait analyses (Calus and Veerkamp, 2011; Jia and Jannink, 2012), where multiple traits are fitted in the model simultaneously. Another approach to borrow information from other sources of data is the use of "single-step" methods described below.

The prediction methods described above use only phenotypic information from individuals with genomic information. In general, the number of individuals with genomic information is

a small subset of the individuals represented in the population with pedigree and phenotypic information. Thus, single-step methodologies were developed to take advantage of all pedigree, phenotypic and genomic information simultaneously (Legarra et al., 2009; Fernando et al., 2014). In what is known as single-step genomic BLUP (SSGBLUP), an ingenious strategy is used to construct a relationship matrix that combines genotypic and pedigree information. These single-step methodologies were shown to yield a similar or higher accuracy for genotyped individuals (Misztal et al., 2013; Lourenco et al., 2014, 2015) by borrowing information from other non-genotyped individuals compared to methods using only genotyped individuals. Fernando et al. (Fernando et al., 2014) proposed a class of single-step Bayesian regression methods (SSBR) to extend SSGBLUP to incorporate BayesB-like or BayesC-like models for SNP effects. SSBR methods may promise higher prediction accuracies and provide computational benefits when many animals are genotyped.

1.1.3 Validation strategies

Cross validation is often used to quantify the predictive ability of a statistical model, and it is used routinely to test prediction performance in animal and plant breeding (Gianola and Rosa, 2015). For example, in genomic prediction, the dataset is split into two partitions, a training set and a testing set, and all the marker effects are estimated simultaneously using the data from training set. Then, these estimates are used to predict breeding values of individuals in the testing set. In k-fold cross validation (Hastie et al., 2009), the whole dataset is partitioned into k parts with k analyses, where one part is omitted for training with validation on the omitted part. Leave-one-out cross validation (LOOCV) is a special case of k-fold cross validation with $k = n$, the number of observations. When the dataset is small, leave-one-out cross validation is appealing as the size of the training set is maximized. However, naive application of LOOCV is computationally intensive, requiring n analyses.

1.2 Thesis Organization

The objective of this thesis is to propose or investigate statistical models and computational algorithms to improve the prediction accuracy or computational efficiency of the whole genome

analyses. In this thesis, several statistical models and computational algorithms were proposed and investigated, contributing to the three areas of research and development in whole genome analyses described above. The proposed methods improved either the prediction accuracy or the computational efficiency of the analyses.

1.2.1 Data simulation

In Chapter 2, a simulation strategy is described to drop down origins and positions of chromosomal segments rather than every allele state to efficiently simulate sequence data and complex pedigree structures across multiple generations. A software tool XSim, which incorporates the efficient strategy, has been developed with implementations in C++ (Stroustrup, 2013) and Julia (Bezanson et al., 2017). XSim allows the genome of founders to be characterized by real genome sequence data and complex pedigree structures among descendants.

1.2.2 Use of genomic data for prediction and GWAS

Statistical models and computational algorithms were considered in chapter 3-6 to improve either the prediction accuracy or the computational efficiency of whole genome analyses.

1.2.2.1 computational efficiency

In Chapter 3, we showed how Gibbs samplers without the Metropolis-Hastings algorithm can be used for the BayesB method. We showed that by introducing a variable δ_j in BayesB, indicating whether the marker effect for a locus is zero or non-zero, the marker effect and locus-specific variance can be sampled using Gibbs. We considered three different versions of the Gibbs sampler to sample each marker effect, locus-specific variance and its indicator variable δ_j. The performance of these samplers were studied.

In Chapter 4, an strategy to parallelize the Gibbs sampling for each marker within each step of the Markov chain Monte Carlo is shown for the BayesC prior, using an augmented marker covariate matrix. The use of this strategy is expected to increase the speed of Gibbs sampling with the use of less memory. Use of this approach with other priors, such as those in BayesA, BayesB, should be straightforward.

1.2.2.2 accuracy

In Chapter 5, we proposed a flexible variable selection model for multiple-trait analyses with BayesCπ or BayesB priors. A previous proposed multi-trait BayesCπ methods presented by Jia et al. (Jia and Jannink, 2012) assumes a locus affects none of the traits or has simultaneous effects on all traits. Our model, however, allows loci to have effects on any number of traits. Our new methods were compared to this previously used multi-trait method and single-trait methods using real and simulated data.

In Chapter 6, alternative approaches to single-trait genomic prediction using genotyped and non-genotyped Hanwoo beef cattle were compared. The single-step Bayesian regression method was compared to single-step genomic best linear unbiased prediction method and several other methods using only pedigree or genotypic information, such as pedigree-based BLUP, BayesB and BayesC.

1.2.3 Validation strategies

In Chapter 7, efficient LOOCV strategies were proposed for genomic BLUP in scenarios when $n > p$ or $n < p$. These strategies were compared to naive application of LOOCV with simulated data.

Chapter 8 concludes this thesis.

CHAPTER 2. XSIM: SIMULATION OF DESCENDANTS FROM ANCESTORS WITH SEQUENCE DATA

Hao Cheng, Dorian Garrick and Rohan Fernando

A paper published in G3: Genes, Genomes, Genetics

2.1 Abstract

Real or imputed high-density SNP genotypes are routinely used for genomic prediction and genome-wide association studies. Many researchers are moving towards the use of actual or imputed next generation sequence data in whole-genome analyses. Simulation studies are useful to mimic complex scenarios and test different analytical methods. We have developed a software tool XSim to efficiently simulate sequence data in descendants in arbitrary pedigrees. In this software, a strategy to drop down origins and positions of chromosomal segments rather than every allele state is implemented to simulate sequence data and accommodate complicated pedigree structures across multiple generations. Both C++ and Julia versions of XSim have been developed.

2.2 Introduction

Analysis of real or imputed genotypes for genomic prediction and genome-wide association studies can result in findings that are difficult to validate. Simulated data have advantages in that the underlying causal mutations and simulated breeding values are available for direct validation. In general, there are two types of simulation methods: coalescent methods and forward-in-time (drop down) methods. Compared to coalescent-based simulations, forward-in-time simulations are very flexible, which allows modeling large numbers of recombination

events in concert with complex life-like selection scenarios (Chadeau-Hyam et al., 2008; Hoggart et al., 2007). However, forward-in-time methods, which drop allele states down the pedigree to simulate and record genomic information for every individual in the entire population, are computationally intensive (Hoggart et al., 2007). Here a strategy is described to drop down origins and positions of chromosomal segments rather than every allele state to efficiently simulate sequence data and complicated pedigree structures across multiple generations. A software tool XSim, which incorporates our efficient strategy, has been developed to use founders characterized by real genome sequence data and complicated pedigree structures among descendants.

2.3 Materials and Methods

2.3.1 Simulation method

The basic idea of our strategy is to record the starting positions and founder origins (founder chromosome identifiers) of each chromosome segment rather than the allele state at each locus for the whole genome.

At first, entire chromosomes in founders are labeled with unique identifiers. Without considering mutations, each chromosome in each descendant individual can be represented using a pair of vectors: a vector of crossover positions and a vector of founder origins. In addition to the position and origin vectors, the allele states of the founder genomes need to be either generated from user-defined map positions and allele frequencies or obtained from real haplotypes or sequence data. As explained in the example below, during meiosis, the gamete that is formed will contain chromosomal segments from the paternal and maternal chromosomes of the parent with new segments introduced on either side of any crossover sites.

An example to illustrate the simulation strategy is shown in figure 1. The pairs of starting base pair position and origin vectors for founder 1 are {[0], [a]} and {[0], [b]}. During meiosis, assuming a crossover occurs at base pair position 12 (e.g. 12.0 Mb), the pair of position and origin vectors for one of the resulting recombinant chromosomes is {[0, 12], [a, b]}. Similarly, pairs of position and origin vectors for founder 2 are {[0], [c]} and {[0], [d]}. Assuming a crossover occurs at base pair position 47 (e.g. 47.0 Mb), the pair of position and origin vectors

for one of the resulting recombinant chromosomes is {[0, 47], [c, d]} . Thus, the chromosomes of the offspring of founder 1 and founder 2 are {[0, 12], [a, b]} and {[0, 47], [c, d]}. Suppose a crossover occurs at base pair position 32 during meiosis in this offspring. Then, the pair of position and origin vectors for one of the resulting recombinant chromosomes is {[0, 12, 32, 47], [a, b, c, d]}. Then, given the positions of the crossover sites and corresponding origins of chromosomes, the entire genome of any non-founder can be constructed to the density of the founder genomes. In the classical gene drop method, all allele states are dropped down sequentially from founders all the way to last generation. However, in the drop-down strategy proposed here, what are dropped down sequentially from founders to the last generation are sparse vectors containing only founder origins and crossover positions. Thus, in the absence of mutation, computing time and memory requirement to drop down this genomic information over generations is free of the number of loci. Once the origin and position vectors are available in the latter generations of interest, allele state information from founders can be dropped down directly to this generation.

As one can observe from the example, in each meiosis, position and origin vectors will grow in size due to new crossover sites. When the paternal and maternal chromosome segments have the same founder origins at the crossover site, the position of the crossover site is not recorded in the resultant recombinant chromosomes. The probability of this happening is inversely related to the effective population size. Sometimes crossover events will result in reducing the length of the position and origin vectors. It can be observed that the length of these two vectors plateaus to a constant that depends on the effective population size (Goddard, 2008). Thus, when the effective population size is approximately 100, for example, and the number of loci being simulated on each chromosome is more than 1,000, our simulation method will be much faster than the classical gene drop method, which sequentially simulates the passage of all allele states from founders to last generation.

Besides crossover positions and founder origins, mutations can be tracked by recording in an additional vector the positions of inherited and de novo mutation sites for each chromosome in each individual. The growth rate of the "mutation" vectors depends on the mutation rate and the number of loci being simulated. Unlike the position and origin vectors, the length of

mutation vectors keeps growing. When the length of the mutation vectors becomes too long for efficient computation, allele states from founders can be dropped down directly to current non-founders. Then these non-founders can be relabelled and therefore treated as new founders. Now, their chromosomes will be labeled with unique identifiers, and this reduces the length of origin and position vectors to one. For these individuals, the length of mutation vectors will be reduced to zero. This strategy has been adopted in XSim.

In summary, three vectors are used to represent each chromosome in each non-founder: the first vector to record crossover positions, the second to record origins of chromosomes and the third to record mutation sites. This strategy is efficient because these vectors are sparse relative to the allele state vectors used in classical gene drop approach.

2.3.2 Software tool

In the C++ software tool, three hierarchical C++ classes referred to as LocusInfo, ChromosomeInfo and GenomeInfo were defined to specify genetic characteristics at locus, chromosome and genome levels. These classes can be used to specify user-defined parameters such as allele frequencies, map positions, number of loci, chromosome lengths, numbers of chromosomes and mutation rates. Values for these parameters can also be generated randomly.

Three C++ classes Animal, Cohort and Population are defined to simulate the passage of the information on collections of individuals over generations. Complex mating structures such as cross breeding, overlapping generations and arbitrary user-defined pedigrees are straightforward. In XSim, real haplotype data such as from the 1,000 human genomes project can be used for founders rather than limited user-defined parameters.

XSim has also been implemented in Julia, a new dynamic programming language. The performance of Julia is often similar to that of C++. Compared to C++, software tools written in Julia are more user-friendly.

Both C++ and Julia versions of XSim are available with source code from http://QTL.rocks.

2.4 Discussion

This paper describes an efficient strategy to simulate descendants forward in time from ancestors with any density of variant information up to and including sequence data, which can be obtained for founders by sequencing or simulation. This strategy has been implemented in both C++ and Julia versions of XSim. The software tool XSim incorporating this efficient strategy has been developed to use founders characterized by any density of variant information and complicated pedigree structures among descendants.

Several forward-in-time simulation packages have been developed to simulate sequence data. Similar strategies to drop down origins and positions of chromosomal segments have been described (Haiminen et al., 2013; Aberer and Stamatakis, 2013; Kessner and Novembre, 2014) and implemented in packages such as forqs (Kessner and Novembre, 2014). These packages, however, are not developed to accommodate simulations from ancestors with real sequence data. In forqs, the growth of position and origin vectors are considered to be linear with number of generations due to recombination. However, in our simulation strategy, the length of position and origin vectors plateaus to a constant. Fregene is also an efficient forward-in-time simulation package. It is efficient when all loci in founders are homozygous. Fregene, however, does not accommodate simulations from ancestors with real sequence data (Hoggart et al., 2007).

Author's contributions

DG, RF, HC contributed to the development of the methods. HC and RF wrote the program code. The manuscript was prepared by RF and HC. All authors read and approved the final manuscript.

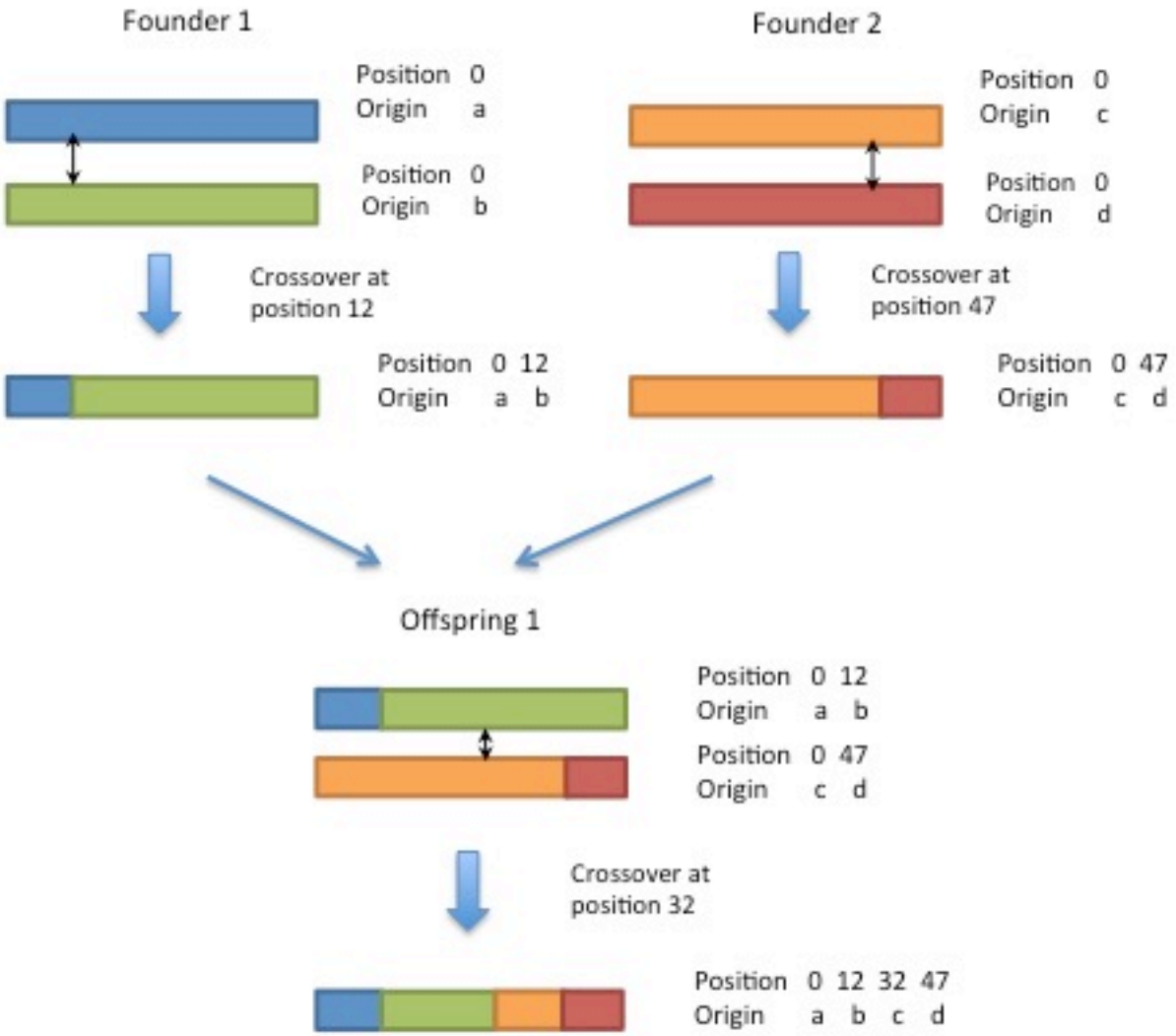

Figure 1: An example to illustrate the simulation strategy
(crossover sites indicated by ↕)

Figure 2.1 An example to illustrate the simulation strategy

CHAPTER 3. A FAST AND EFFICIENT GIBBS SAMPLER FOR BAYESB IN WHOLE GENOME ANALYSES

Hao Cheng, Long Qu, Dorian Garrick and Rohan Fernando

A paper published in Genetics Selection Evolution

3.1 Abstract

In whole-genome analyses, the number p of marker covariates is often much larger than the number n of observations. Bayesian multiple regression models are widely used in genomic selection to address this problem of $p \gg n$. The primary difference between these models is the prior assumed for the effects of the covariates. Usually in the BayesB method, a Metropolis-Hastings (MH) algorithm is used to jointly sample the marker effect and the locus-specific variance, which may make BayesB computationally intensive. In this paper we show how the Gibbs sampler without the MH algorithm can be used for the BayesB method. We consider three different versions of the Gibbs sampler to sample the marker effect and locus-specific variance for each locus. Among the Gibbs samplers that were considered, the most efficient sampler is about 2.1 times as efficient as the MH algorithm proposed by Meuwissen et al. and 1.7 times as efficient as that proposed by Habier et al. The three Gibbs samplers were twice as efficient as Metropolis-Hastings samplers and gave virtually the same results.

3.2 Introduction

In whole-genome analyses, the number p of marker covariates is often much larger than the number n of observations. Bayesian multiple regression models are widely used in genomic selection to address this problem of $p \gg n$. The primary difference between these models is the

prior assumed for the effects of the covariates. These priors and their effects on inference have been recently reviewed by Gianola (Gianola, 2013). In most Bayesian analyses of whole-genome data, inferences are based on Markov chains constructed to have a stationary distribution equal to the posterior distribution of the unknown parameters of interest (Norris, 1997). This is often done by employing a Gibbs sampler where samples are drawn from the full-conditional distributions of the parameters (Sorensen and Gianola, 2002).

It can be shown that in BayesA introduced by Meuwissen et al. (Meuwissen et al., 2001a), the prior for each marker effect follows a scaled t distribution (Gianola et al., 2009a). However when the prior for the marker effect is specified as a t distribution, its full-conditional is not of a known form. Fortunately, this prior can also be specified as a normally distributed marker effect conditional on a locus specific variance, which is given a scaled inverted chi-square distribution. When marginalized over the variance, this gives a t distribution for the marker effect (Gianola et al., 2009a). Thus, the posterior for the marker effect would be identical under both these priors. The second form of the prior, however, is more convenient because it results in the full-conditional for the marker effect having a normal distribution.

BayesA is a special case of BayesB, also introduced by Meuwissen et al. (Meuwissen et al., 2001a), where the prior for each marker effect follows a mixture distribution with a point mass at zero with probability π and a univariate-t distribution with probability $1-\pi$ (Gianola et al., 2009a). When $\pi = 0$ BayesB becomes BayesA. When the marker effect is non-null, as in BayesA, the second form of the prior leads to the full-conditional of the marker effect being normal. Nevertheless, Meuwissen et al. (Meuwissen et al., 2001a) used a Metropolis-Hastings (MH) algorithm to jointly sample the marker effect and the locus-specific variance because they argued that "the Gibbs sampler will not move through the entire sampling space" for BayesB. In their MH algorithm, they use the prior distribution of the locus-specific variance as the proposal distribution. When π is high, the proposed values for the marker effect will be zero with high probability. Thus, for each locus, 100 cycles of MH algorithm were used in their paper, which makes BayesB computationally intensive. Habier et al. (Habier et al., 2011a) used an alternative proposal, where the marker effect was zero with probability 0.5, that leads to a more efficient MH algorithm. For each locus, 5 cycles of MH were used to sample marker

effects in this efficient MH method.

In this paper we will show how Gibbs samplers without the MH algorithm can be used for the BayesB method. Recall that by introducing a locus-specific variance into BayesA, the full-conditional for the marker effects becomes normal. Similarly, in this paper we show that by introducing a variable δ_j in BayesB, indicating whether the marker effect for a locus is zero or non-zero, the marker effect and locus-specific variance can be sampled using Gibbs. We consider three different versions of the Gibbs sampler to sample each marker effect, locus-specific variance and its indicator variable δ_j. The objectives of this paper are to introduce these samplers and study their performance.

3.3 Materials and Methods

BayesB introduced by Meuwissen et al. (Meuwissen et al., 2001a) assumes each locus-specific variance follows a mixture distribution. However, following Gianola (Gianola et al., 2009a), we prefer to specify the mixture at the level of the marker effect instead of the locus specific variance. In this formulation, the prior for the marker effect is a mixture with a point mass at zero and a univariate normal distribution conditional on σ_j^2:

$$\left(\alpha_j|\sigma_j^2\right)\begin{cases} = 0 & \text{with probability } \pi \\ \sim N\left(0,\sigma_j^2\right) & \text{with probability } (1-\pi)\,, \end{cases} \tag{3.1}$$

where σ_j^2 follows a scaled inverted chi-square distribution and π is treated as known. Employing the concept of data augmentation, it is convenient to write the marker effect α_j as $\alpha_j = \beta_j\delta_j$, where we introduce a Bernoulli variable δ_j with probability of success $1-\pi$ and normally distributed variable β_j with mean zero and variance σ_j^2, which has a scaled inverted chi-square distribution. As shown below, Gibbs sampling can be used to draw samples for these unknowns.

3.3.1 Gibbs Samplers for BayesB

Here we present three Gibbs samplers for BayesB. The first is a single-site Gibbs sampler, where all parameters are sampled from their full conditional distributions. The second is a blocking Gibbs sampler, where δ_j, β_j are sampled from the joint full-conditional distribution

$f\left(\delta_j, \beta_j | \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right)$, where $\boldsymbol{\xi} = \left[\sigma_1^2, \sigma_2^2, \ldots, \sigma_k^2\right]$, because δ_j and β_j are highly dependent. Carlin and Chib have shown that the prior used for parameters that are not in the model does not affect the Bayes factor (Carlin and Chib, 1995). Thus, this prior, which they call a pseudo prior, can be chosen to improve mixing of the sampler. Following Carlin and Chib, the third sampler is a Gibbs sampler where a pseudo prior is used for β_j when δ_j is zero. Godsill has shown that the marginal posterior for a parameter in the model does not depend on the choice of pseudo priors (Godsill, 2001). It has been suggested to choose the full conditional distribution for β_j when it is in the model as the pseudo prior (Carlin and Chib, 1995; Godsill, 2001). This choice is justified by showing that using this pseudo prior is equivalent to sampling δ_j from $f\left(\delta_j | \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right)$ (Godsill, 2001). However, in BayesB, use of the exact full conditional distribution as the pseudo prior will require MH to sample σ_j^2 and σ_e^2. Thus in this paper, a distribution close to the full conditional is employed.

3.3.2 BayesB model with data augmentation

3.3.2.1 Model

$$y_i = \mu + \sum_{j=1}^{k} X_{ij}\beta_j\delta_j + e_i \tag{3.2}$$

where y_i is the phenotype for individual i, μ is the overall mean, k is the number of SNPs, X_{ij} is the genotype covariate at locus j for animal i (coded as 0, 1, 2), β_j is the allele substitution effect for locus j, δ_j is an indicator variable and e_i is the random residual effect for individual i.

3.3.2.2 Priors

The prior for μ is a constant. The prior for e_i is $e_i|\sigma_e^2 \overset{iid}{\sim} N\left(0, \sigma_e^2\right)$ and $\left(\sigma_e^2|\nu_e, S_e^2\right) \sim \nu_e S_e^2 \chi_{\nu_e}^{-2}$. The prior for β_j is $\beta_j|\sigma_j^2 \sim N\left(0, \sigma_j^2\right)$. The prior for $\left(\sigma_j^2|\nu_\beta, S_\beta^2\right) \sim \nu_\beta S_\beta^2 \chi_{\nu_\beta}^{-2}$, The prior for δ_j is

$$(\delta_j|\pi)\begin{cases}=1 & probability\,(1-\pi)\\ =0 & probability\,\pi.\end{cases}$$

3.3.2.3 Single-site Gibbs Sampler

The full conditional distributions of μ, σ_j^2 and σ_e^2 are well-known (Meuwissen et al., 2001a; Habier et al., 2010a; Fernando and Garrick, 2013a). Thus they are presented here without derivations. The full conditional of μ is a normal distribution with mean $\frac{1'\mathbf{e}}{n}$ and variance $\frac{\sigma_e^2}{n}$, where $e = \mathbf{y} - \sum_{j=1}^{k} \mathbf{X}_j\beta_j\delta_j$ and n is the number of individuals. The full conditional distributions of σ_j^2 and σ_e^2 are both scaled inverted chi-square distributions; for σ_j^2, the scale parameter is $\frac{\left(\nu_\beta S_\beta^2+\beta_j^2\right)}{\nu_\beta+1}$ and the degrees of freedom parameter is $\nu_\beta + 1$; for σ_e^2, the scale parameter is $\frac{\nu_e S_e^2+\boldsymbol{e}'\boldsymbol{e}}{\nu_e+n}$and the degrees of freedom parameter is $\nu_e + n$, where $\mathbf{e} = \mathbf{y} - \mathbf{1}\mu - \sum_{j=1}^{k} \mathbf{X}_j\beta_j\delta_j$.

Next we derive the full conditional distributions of β_j and δ_j. These full conditional distributions are proportional to the joint distribution of all parameters and $\mathbf{y}$, which can be written as

$$\begin{aligned}
f\left(\boldsymbol{y},\mu,\boldsymbol{\beta},\boldsymbol{\delta},\boldsymbol{\xi},\sigma_e^2\right) &\propto f\left(\boldsymbol{y}\mid\mu,\boldsymbol{\beta},\delta,\boldsymbol{\xi},\sigma_e^2\right) f\left(\boldsymbol{\beta}|\boldsymbol{\xi}\right) f\left(\boldsymbol{\delta}\right) f\left(\boldsymbol{\xi}\right) f\left(\sigma_e^2\right)\\
&\propto \left(\sigma_e^2\right)^{-\frac{n}{2}} \exp\left[-\frac{\left(\boldsymbol{y}-\mathbf{1}\mu-\sum_{j=1}^{k}\mathbf{X_j}\beta_j\delta_j\right)'\left(\boldsymbol{y}-\mathbf{1}\mu-\sum_{j=1}^{k}\mathbf{X_j}\beta_j\delta_j\right)}{2\sigma_e^2}\right]\\
&\times\prod_{j=1}^{k}\frac{1}{\sqrt{2\pi}\left(\sigma_j^2\right)^{\frac{1}{2}}}\exp\left(-\frac{\beta_j^2}{2\sigma_j^2}\right)\\
&\times\prod_{j=1}^{k}\pi^{(1-\delta_j)}\left(1-\pi\right)^{\delta_j}\\
&\times\prod_{j=1}^{k}\frac{\left(S_\beta^2\frac{\nu_\beta}{2}\right)^{\frac{\nu_\beta}{2}}}{\Gamma\left(\frac{\nu_\beta}{2}\right)}\left(\sigma_j^2\right)^{\left(-\frac{\nu_\beta+2}{2}\right)}\exp\left(-\frac{\nu_\beta S_\beta^2}{2\sigma_j^2}\right)\\
&\times\frac{\left(S_e^2\frac{\nu_e}{2}\right)^{\frac{\nu_e}{2}}}{\Gamma\left(\frac{\nu_e}{2}\right)}\left(\sigma_e^2\right)^{\left(-\frac{\nu_e+2}{2}\right)}\exp\left(-\frac{\nu_e S_e^2}{2\sigma_e^2}\right). \qquad (3.3)
\end{aligned}$$

The full conditional distribution of β_j is now obtained by dropping factors that do not involve

β_j, which gives

$$f\left(\beta_j|\boldsymbol{y},\mu,\beta_{-j},\boldsymbol{\delta},\boldsymbol{\xi},\sigma_e^2\right) \propto f\left(\boldsymbol{y}\mid\mu,\boldsymbol{\beta},\boldsymbol{\delta},\boldsymbol{\xi},\sigma_e^2\right) f\left(\beta_j|\sigma_j^2\right)$$
$$\propto \exp\left[-\frac{\left(\boldsymbol{y}-\mathbf{1}\mu-\sum_{j=1}^{k}\boldsymbol{X_j}\beta_j\delta_j\right)'\left(\boldsymbol{y}-\mathbf{1}\mu-\sum_{j=1}^{k}\boldsymbol{X_j}\beta_j\delta_j\right)}{2\sigma_e^2}\right]\exp\left(-\frac{\beta_j^2}{2\sigma_j^2}\right)$$
$$\propto \exp\left[-\frac{\left(\boldsymbol{w}-\boldsymbol{X_j}\beta_j\delta_j\right)'\left(\boldsymbol{w}-\boldsymbol{X_j}\beta_j\delta_j\right)}{2\sigma_e^2}\right]\exp\left(-\frac{\beta_j^2}{2\sigma_j^2}\right),$$

where $\boldsymbol{w}=\boldsymbol{y}-\mathbf{1}\mu-\sum_{j'\neq j}\boldsymbol{X}_{j'}\beta_{j'}\delta_{j'}$. When $\delta_j=1$,

$$f\left(\beta_j|\boldsymbol{y},\mu,\beta_{-j},\boldsymbol{\delta},\boldsymbol{\xi},\sigma_e^2\right) \propto \exp\left[-\frac{\left(\boldsymbol{w}-X_j\beta_j\right)'\left(\boldsymbol{w}-X_j\beta_j\right)}{2\sigma_e^2}\right]\exp\left(-\frac{\beta_j^2}{2\sigma_j^2}\right)$$
$$\propto \exp\left[-\frac{1}{2}\frac{\left(\beta_j-\frac{\mathbf{x}_j'\boldsymbol{w}}{c_j}\right)^2}{\frac{\sigma_e^2}{c_j}}\right] \tag{3.4}$$

where $c_j=\boldsymbol{X_j'X_j}+\frac{\sigma_e^2}{\sigma_j^2}$. Now, (3.4) can be recognized as the kernel of a normal distribution with mean $\frac{\mathbf{x}_j'\boldsymbol{w}}{c_j}$ and variance $\frac{\sigma_e^2}{c_j}$. When $\delta_j=0$,

$$f\left(\beta_j|\boldsymbol{y},\mu,\beta_{-j},\boldsymbol{\delta},\boldsymbol{\xi},\sigma_e^2\right) \propto \exp\left(-\frac{w'w}{2\sigma_e^2}\right)\exp\left(-\frac{\beta_j^2}{2\sigma_j^2}\right),$$

and dropping the factor $\left[\exp\left(-\frac{w'w}{2\sigma_e^2}\right)\right]$, which is free of β_j, gives

$$f\left(\beta_j|\boldsymbol{y},\mu,\beta_{-j},\delta,\xi,\sigma_e^2\right) \propto \exp\left(-\frac{\beta_j^2}{2\sigma_j^2}\right),$$

which is the kernel of a normal distribution with null mean and variance σ_j^2. Thus,

$$p(\beta_j|ELSE)=\begin{cases}\sim N\left(\frac{\mathbf{x}_j'\boldsymbol{w}}{c_j},\frac{\sigma_e^2}{c_j}\right) & when\ \delta_j=1,\\ \sim N\left(0,\sigma_j^2\right) & when\ \delta_j=0,\end{cases}$$

where ELSE stands for all the other parameters and y. This means when $\delta_j=1$, the sampling of β_j is identical to that in BayesA; when $\delta_j=0$, β_j is sampled from its prior.

Similarly, the full conditional distribution of δ_j can be obtained from (3.3) by dropping all factors free of δ_j, which gives

$$\begin{aligned}\Pr\left(\delta_j = 1 \mid ELSE\right) &\propto f\left(\boldsymbol{y} \mid \mu, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}, \sigma_e^2\right) \Pr\left(\delta_j = 1\right) \\ &\propto \exp\left[-\frac{\left(\boldsymbol{w} - \boldsymbol{X_j}\beta_j\delta_j\right)'\left(\boldsymbol{w} - \boldsymbol{X_j}\beta_j\delta_j\right)}{2\sigma_e^2}\right](1-\pi) \\ &\propto \exp\left[-\frac{\left(\boldsymbol{w} - X_j\beta_j\right)'\left(\boldsymbol{w} - X_j\beta_j\right)}{2\sigma_e^2}\right](1-\pi),\end{aligned}$$

$$\begin{aligned}\Pr\left(\delta_j = 0 \mid ELSE\right) &\propto f\left(\boldsymbol{y} \mid \mu, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}, \sigma_e^2\right) \Pr\left(\delta_j = 0\right) \\ &\propto \exp\left[-\frac{\left(\boldsymbol{w} - \boldsymbol{X_j}\beta_j\delta_j\right)'\left(\boldsymbol{w} - \boldsymbol{X_j}\beta_j\delta_j\right)}{2\sigma_e^2}\right]\pi \\ &\propto \exp\left(-\frac{\boldsymbol{w}'\boldsymbol{w}}{2\sigma_e^2}\right)\pi.\end{aligned}$$

Thus,

$$\Pr\left(\delta_j = 1 \mid ELSE\right) = \frac{(1-\pi)\exp\left[-\frac{\left(\boldsymbol{w}-\boldsymbol{X_j}\beta_j\right)'\left(\boldsymbol{w}-\boldsymbol{X}_j\beta_j\right)}{2\sigma_e^2}\right]}{(1-\pi)\exp\left[-\frac{\left(\boldsymbol{w}-\boldsymbol{X_j}\beta_j\right)'\left(\boldsymbol{w}-\boldsymbol{X}_j\beta_j\right)}{2\sigma_e^2}\right] + \pi\exp\left(-\frac{\boldsymbol{w}'\boldsymbol{w}}{2\sigma_e^2}\right)},$$

with $\Pr\left(\delta_j = 0 \mid ELSE\right) = 1 - \Pr\left(\delta_j = 1 \mid ELSE\right)$.

3.3.2.4 Joint Gibbs Sampler

The same priors as in single-site Gibbs sampler are used here. The only difference is that δ_j, β_j are sampled from their joint full conditional distribution, which can be written as the product of the full conditional distribution of β_j given δ_j and marginal full conditional distribution of δ_j:

$$f\left(\delta_j, \beta_j | \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right) = f\left(\beta_j | \delta_j, \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right) \Pr\left(\delta_j | \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right).$$

Thus δ_j is first sampled from $f\left(\delta_j | \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right)$. Then β_j is sampled from $f\left(\beta_j | \delta_j, \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right)$, which is identical to the sampling of β_j in BayesB with single-site

Gibbs sampler. The marginal full conditional for δ_j can be written as

$$f\left(\delta_j|\boldsymbol{y},\mu,\boldsymbol{\beta}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\xi},\sigma_e^2\right) \propto f\left(\boldsymbol{y} \mid \delta_j,\mu,\boldsymbol{\beta_{-j}},\boldsymbol{\delta_{-j}},\boldsymbol{\xi},\sigma_e^2\right)\Pr\left(\delta_j\right) \tag{3.5}$$
$$\propto f\left(\boldsymbol{w} \mid \delta_j,\sigma_j^2,\sigma_e^2\right)\Pr\left(\delta_j\right),$$

where $\boldsymbol{w}=\boldsymbol{y}-\mathbf{1}\mu-\sum_{j'\neq j}\boldsymbol{X}_{j'}\beta_{j'}\delta_{j'}=\boldsymbol{e}+\mathbf{X}_j\beta_j\delta_j$. Now $f\left(\boldsymbol{w} \mid \delta_j,\sigma_j^2,\sigma_e^2\right)$ is a multivariate normal distribution with mean $E\left(\boldsymbol{e}+\boldsymbol{X}_j\beta_j\delta_j \mid \delta_j,\sigma_j^2,\sigma_e^2\right)$ and variance $Var\left(\boldsymbol{X}_j\beta_j\delta_j+\boldsymbol{e} \mid \delta_j,\sigma_j^2,\sigma_e^2\right)$. When $\delta_j=1$, it becomes a multivariate normal with null mean and variance $X_jX_j^{'}\sigma_j^2+\sigma_e^2$; when $\delta_j=0$, it becomes a multivariate normal with null mean and variance $\boldsymbol{I}\sigma_e^2$. Thus, samples can be drawn using

$$f\left(\delta_j=1|\boldsymbol{y},\mu,\boldsymbol{\beta}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\xi},\sigma_e^2\right)=\frac{h_1\Pr\left(\delta_j=1\right)}{h_1\Pr\left(\delta_j=1\right)+h_0\Pr\left(\delta_j=0\right)}, \tag{3.6}$$
$$=\frac{1}{1+\frac{h_0Pr(\delta_j=0)}{h_1\Pr(\delta_j=1)}} \tag{3.7}$$

where $h_i=f\left(\boldsymbol{w} \mid \delta_j=i,\sigma_j^2,\sigma_e^2\right)$. However, evaluating the multivariate normal distribution $f\left(\boldsymbol{w} \mid \delta_j,\sigma_j^2,\sigma_e^2\right)$ is computationally intense. An efficient way is to use the univariate distribution of $\boldsymbol{X}_j^{'}\boldsymbol{w}$, which contains all the information from $\boldsymbol{w}$ about β_j, instead of the distribution of $\boldsymbol{w}$, which is a multivariate. Thus, (3.6) can be written as

$$f\left(\delta_j=1|\boldsymbol{y},\mu,\boldsymbol{\beta}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\xi},\sigma_e^2\right)=\frac{m_1\Pr\left(\delta_j=1\right)}{m_1\Pr\left(\delta_j=1\right)+m_0\Pr\left(\delta_j=0\right)},$$

where $m_i=f\left(\boldsymbol{X_j'w} \mid \delta_j=i,\mu,\boldsymbol{\beta}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\xi},\sigma_e^2\right)$, m_1 is an univariate normal distribution with null mean and variance $\left(\boldsymbol{X_j'X_j}\right)^2\sigma_j^2+\boldsymbol{X_j'X_j}\sigma_e^2$, and m_0 is an univariate normal distribution with null mean and variance $\boldsymbol{X_j'X_j}\sigma_e^2$.

3.3.2.5 Gibbs Sampler with pseudo priors

Here, following Carlin and Chib(Carlin and Chib, 1995), a pseudo prior is used for β_j when δ_j is zero. They proposed to use the full conditional distribution of β_j when $\delta_j=1$ as the pseudo prior for β_j when $\delta_j=0$ (Carlin and Chib, 1995; Godsill, 2001), which results in the prior for β_j as

$$\beta_j|\delta_j = \begin{cases} \sim N\left(0, \sigma_j^2\right) & when\ \delta_j = 1 \\ \sim f\left(\beta_j|\delta_j = 1, \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right) & when\ \delta_j = 0. \end{cases}$$

We show below that the posterior mean of the marker effect, $\alpha_j = \beta_j\delta_j$, does not depend on the pseudo prior. This posterior mean can be written as

$$\begin{aligned} & E\left(\beta_j\delta_j \mid \mathbf{y}\right) \\ =& \sum_{\delta_j} \int \beta_j\delta_j f\left(\beta_j, \delta_j \mid \boldsymbol{y}\right) d\beta_j \\ =& \frac{\sum_{\delta_j} \int \beta_j\delta_j f\left(\boldsymbol{y} \mid \beta_j, \delta_j\right) f\left(\beta_j \mid \delta_j\right) \Pr\left(\delta_j\right) d\beta_j}{f\left(y\right)}. \end{aligned} \tag{3.8}$$

The numerator in (3.8) is

$$\begin{aligned} & \int \beta_j f\left(\boldsymbol{y} \mid \beta_j, \delta_j = 1\right) f\left(\beta_j \mid \delta_j = 1\right) \Pr\left(\delta_j = 1\right) d\beta_j \\ +& \int 0 f\left(\boldsymbol{y} \mid \beta_j, \delta_j = 0\right) f\left(\beta_j \mid \delta_j = 0\right) \Pr\left(\delta_j = 0\right) d\beta_j \\ =& \int \beta_j f\left(\boldsymbol{y} \mid \beta_j, \delta_j = 1\right) f\left(\beta_j \mid \delta_j = 1\right) \Pr\left(\delta_j = 1\right) d\beta_j, \end{aligned}$$

which is free of the pseudo prior: $f\left(\beta_j \mid \delta_j = 0\right)$. Further, it can be seen from the model equation (3.2) that the value of $\mathbf{y}$ is free of β_j when $\delta_j = 0$. Thus, the marginal distribution of $\mathbf{y}$, the denominator of (3.8), does not depend on the pseudo prior, which is the distribution of β_j when $\delta_j = 0$. As both the numerator and denominator of (3.8) are free of the pseudo prior, it follows that the posterior mean of α_j does not depend on the pseudo prior for β_j.

We show here that, given this pseudo prior, the full conditional of δ_j is identical to the marginal full conditional distribution of δ_j, $\Pr\left(\delta_j|\boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right)$, which is used in the joint Gibbs sampler.

The full conditional probability of $\delta_j = 1$ can be written as

$$\begin{aligned} \Pr\left(\delta_j = 1 \mid ELSE\right) &= \frac{g_1}{g_1 + g_0} \\ &= \frac{1}{1 + \frac{g_0}{g_1}} \end{aligned} \tag{3.9}$$

where

$$\begin{aligned} g_i &= Pr\left(\delta_j = i \mid ELSE\right) \\ &\propto f\left(\boldsymbol{y} \mid \delta_j = i, \beta_j, \delta_{-j}, \beta_{-j}, \mu, \boldsymbol{\xi}, \sigma_e^2\right) \times f\left(\beta_j \mid \delta_j = i, \sigma_j^2\right) \times p\left(\delta_j = i\right) \\ &\propto f\left(\boldsymbol{w} \mid \delta_j = i, \beta_j, \sigma_e^2\right) \times f\left(\beta_j \mid \delta_j = i, \sigma_j^2\right) \times p\left(\delta_j = i\right) \end{aligned}$$

with $i = 0$ or 1.

The ratio in the denominator of (3.9) is

$$\begin{aligned} \frac{g_0}{g_1} &= \frac{f\left(\boldsymbol{w} \mid \delta_j = 0, \beta_j, \sigma_e^2\right) \times f\left(\beta_j \mid \delta_j = 0\right) \times Pr\left(\delta_j = 0\right)}{f\left(\boldsymbol{w} \mid \delta_j = 1, \beta_j, \sigma_e^2\right) \times f\left(\beta_j \mid \delta_j = 1\right) \times Pr\left(\delta_j = 1\right)} \\ &= f\left(\boldsymbol{w} \mid \delta_j = 0, \beta_j, \sigma_e^2\right) \qquad (3.10) \\ &\times \frac{f\left(\beta_j \mid \delta_j = 0\right)}{f\left(\boldsymbol{w} \mid \delta_j = 1, \beta_j, \sigma_e^2\right) \times f\left(\beta_j \mid \delta_j = 1, \sigma_j^2\right)} \qquad (3.11) \\ &\times \frac{Pr\left(\delta_j = 0\right)}{Pr\left(\delta_j = 1\right)} \end{aligned}$$

In the above equation, (3.10) is identical to h_0 in (3.6) used in the joint Gibbs sampler, because $f\left(\boldsymbol{w} \mid \delta_j = 0, \beta_j, \sigma_e^2\right)$ is also a multivariate normal with null mean and variance $\boldsymbol{I}\sigma_e^2$. We show below that (3.11) is identical to h_1^{-1}. Our proposed prior for β_j when $\delta_j = 0$ (the pseudo prior) can be written as

$$\begin{aligned} & p\left(\beta_j \mid \delta_j = 0\right) \\ =& f\left(\beta_j | \delta_j = 1, \boldsymbol{y}, \mu, \boldsymbol{\beta}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\xi}, \sigma_e^2\right) \\ =& f\left(\beta_j | \delta_j = 1, \boldsymbol{w}, \sigma_e^2, \sigma_j^2\right) \\ =& \frac{f\left(\boldsymbol{w} \mid \delta_j = 1, \beta_j, \sigma_e^2\right) f\left(\beta_j \mid \delta_j = 1, \sigma_j^2\right)}{f\left(\boldsymbol{w} \mid \delta_j = 1, \sigma_j^2, \sigma_e^2\right)}. \end{aligned} \qquad (3.12)$$

After replacing the pseudo prior in (3.11) with (3.12), it becomes

$$\frac{1}{f\left(\boldsymbol{w} \mid \delta_j = 1, \sigma_j^2, \sigma_e^2\right)},$$

which is identical to h_1^{-1}. Thus, the ratio $\frac{g_0}{g_1}$ in (3.9) is identical to $\frac{h_0 Pr(\delta_j=1)}{h_1 Pr(\delta_j=0)}$ in (3.7), which proves the full conditional probability (3.9) of $\delta_j = 1$, when the proposed prior is used, is

identical to (3.7), the marginal full conditional probability of $\delta_j = 1$, which is used in the joint Gibbs sampler.

Use of the exact full conditional distribution as the pseudo prior in BayesB, however, will require MH to sample σ_j^2 and σ_e^2. Thus, a distribution close to the full conditional is employed here. Here we use a normal distribution with mean $\frac{\mathbf{X}_j'\boldsymbol{w}}{X_j'X_j+\widetilde{\lambda}}$and variance $\frac{\widetilde{\sigma_e^2}}{X_j'X_j+\widetilde{\lambda}}$, where $\widetilde{\lambda} = \frac{\widetilde{\sigma_e^2}}{\widetilde{\sigma_j^2}}$, and $\widetilde{\sigma_e^2}$ and $\widetilde{\sigma_j^2}$ are means of the prior distributions for the residual and the marker effect variances respectively.

Next we will show the derivation of the full conditionals, which are proportional to the joint distribution of all parameters and $\mathbf{y}$. Here the joint distribution of all parameters and $\boldsymbol{y}$ can be written as

$$f\left(\boldsymbol{y}, \mu, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}, \sigma_e^2\right) \propto f\left(\boldsymbol{y} \mid \mu, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}, \sigma_e^2\right) f\left(\boldsymbol{\beta} \mid \boldsymbol{\delta}, \boldsymbol{\xi}\right) f\left(\boldsymbol{\delta}\right) f\left(\boldsymbol{\xi}\right) f\left(\sigma_e^2\right) \tag{3.13}$$

$$\propto \left(\sigma_e^2\right)^{-\frac{n}{2}} \exp\left[-\frac{\left(\boldsymbol{y} - \mathbf{1}\mu - \sum_{j=1}^{k} \mathbf{X_j}\beta_j\delta_j\right)'\left(\boldsymbol{y} - \mathbf{1}\mu - \sum_{j=1}^{k} \mathbf{X_j}\beta_j\delta_j\right)}{2\sigma_e^2}\right] \tag{3.14}$$

$$\times \prod_{j=1}^{k}\left[\frac{1}{\sqrt{2\pi}\left(\sigma_j^2\right)^{\frac{1}{2}}} \exp\left(-\frac{\beta_j^2}{2\sigma_j^2}\right)\right]^{\delta_j} \tag{3.15}$$

$$\times \prod_{j=1}^{k}\left\{\frac{1}{\sqrt{2\pi}\left(\frac{\widetilde{\sigma_e^2}}{\boldsymbol{X}_j'\boldsymbol{X}_j+\widetilde{\lambda}}\right)^{\frac{1}{2}}} \exp\left[-\frac{\left(\beta_j - \frac{\boldsymbol{X}_j'\boldsymbol{w}}{\boldsymbol{X}_j'\boldsymbol{X}_j+\widetilde{\lambda}}\right)^2}{2\frac{\widetilde{\sigma_e^2}}{\boldsymbol{X}_j'\boldsymbol{X}_j+\widetilde{\lambda}}}\right]\right\}^{1-\delta_j} \tag{3.16}$$

$$\times \prod_{j=1}^{k} \pi^{(1-\delta_j)}\left(1-\pi\right)^{\delta_j} \tag{3.17}$$

$$\times \prod_{j=1}^{k} \frac{\left(S_\beta^2\frac{\nu_\beta}{2}\right)^{\frac{\nu_\beta}{2}}}{\Gamma\left(\frac{\nu_\beta}{2}\right)}\left(\sigma_j^2\right)^{\left(-\frac{\nu_\beta+2}{2}\right)} \exp\left(-\frac{\nu_\beta S_\beta^2}{2\sigma_j^2}\right) \tag{3.18}$$

$$\times \frac{\left(S_e^2\frac{\nu_e}{2}\right)^{\frac{\nu_e}{2}}}{\Gamma\left(\frac{\nu_e}{2}\right)}\left(\sigma_e^2\right)^{\left(-\frac{\nu_e+2}{2}\right)} \exp\left(-\frac{\nu_e S_e^2}{2\sigma_e^2}\right). \tag{3.19}$$

It's easy to see the full conditional distribution of σ_e^2, which does not involve δ_j, is same as that in the single-site Gibbs sampler. Even though μ also appears in $\boldsymbol{w}$ in (3.16), (3.16) has no effect on the full conditional of μ because the columns of X, which are always centered,

are orthogonal to the column vectors of ones so that $\boldsymbol{X_j'}\mathbf{1}\mu = \mathbf{0}$. Thus, the full conditional of μ is the same as that in the single-site Gibbs sampler. When $\delta_j = 1$, the full conditional distribution of β_j is identical to that in the single-site Gibbs sampler. When $\delta_j = 0$, (3.16) is the only part that includes β_j. Thus the full conditional distribution of β_j is

$$p(\beta_j|ELSE) = \begin{cases} \sim N\left(\frac{\mathbf{X}_j'\boldsymbol{w}}{\boldsymbol{X_j'X_j}+\frac{\sigma_e^2}{\sigma_j^2}}, \frac{\sigma_e^2}{\boldsymbol{X_j'X_j}+\frac{\sigma_e^2}{\sigma_j^2}}\right) & when\ \delta_j = 1, \\ \sim N\left(\frac{\mathbf{X}_j'\boldsymbol{w}}{\boldsymbol{X_j'X_j}+\widetilde{\lambda}}, \frac{\widetilde{\sigma_e^2}}{\boldsymbol{X_j'X_j}+\widetilde{\lambda}}\right) & when\ \delta_j = 0. \end{cases}$$

When $\delta_j = 1$, the full conditional distribution of σ_j^2 is same as that in the single-site Gibbs sampler. When $\delta_j = 0$, (3.18) is the only part that contains σ_j^2, which means it should be sampled from its prior. Thus when $\delta_j = 1$, the full conditional distribution of σ_j^2 is a scaled inverted chi-square distributions with scale parameter $\frac{\nu_\beta S_\beta^2+\beta_j^2}{\nu_\beta+1}$ and degrees of freedom parameter $\nu_\beta + 1$; when $\delta_j = 0$, it is a scaled inverted chi-square distributions with scale parameter S_β^2 and degrees of freedom parameter ν_β. The full conditional distribution of δ_j can be obtained from the joint distribution of all parameters and $\mathbf{y}$ by dropping all factors free of δ_j, which gives

$$\Pr\left(\delta_j = 1 \mid ELSE\right) \propto f\left(\boldsymbol{y} \mid \mu, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}, \sigma_e^2\right) \Pr\left(\delta_j = 1\right) f\left(\beta_j \mid \delta_j = 1\right),$$

and

$$\Pr\left(\delta_j = 0 \mid ELSE\right) \propto f\left(\boldsymbol{y} \mid \mu, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}, \sigma_e^2\right) \Pr\left(\delta_j = 0\right) f\left(\beta_j \mid \delta_j = 0\right).$$

Compared to the full conditional distributions for δ_j in the single-site Gibbs sampler, the difference is the extra factor $f\left(\beta_j \mid \delta_j = 1\right)$ and $f\left(\beta_j \mid \delta_j = 0\right)$, because they cannot be canceled out as in the single-site Gibbs sampler. Thus,

$$\Pr\left(\delta_j = 1 \mid ELSE\right) = \frac{(1-\pi)\exp\left[-\frac{\left(\boldsymbol{w}-\boldsymbol{X_j}\beta_j\right)'\left(\boldsymbol{w}-\boldsymbol{X}_j\beta_j\right)}{2\sigma_e^2}\right] f\left(\beta_j \mid \delta_j = 1\right)}{(1-\pi)\exp\left[-\frac{\left(\boldsymbol{w}-\boldsymbol{X_j}\beta_j\right)'\left(\boldsymbol{w}-\boldsymbol{X}_j\beta_j\right)}{2\sigma_e^2}\right] f\left(\beta_j \mid \delta_j = 1\right) + \pi\exp\left(-\frac{\boldsymbol{w'w}}{2\sigma_e^2}\right) f\left(\beta_j \mid \delta_j = 0\right)},$$

where $f\left(\beta_j \mid \delta_j = 1\right) = \frac{1}{\sqrt{2\pi}\left(\sigma_j^2\right)^{\frac{1}{2}}}\exp\left(-\frac{\beta_j^2}{2\sigma_j^2}\right)$ and $f\left(\beta_j \mid \delta_j = 0\right) = \frac{1}{\sqrt{2\pi}\left(\frac{\widetilde{\sigma_e^2}}{X_j'X_j+\widetilde{\lambda}}\right)^{\frac{1}{2}}}\exp\left[-\frac{\left(\beta_j-\frac{X_j'\boldsymbol{w}}{X_j'X_j+\widetilde{\lambda}}\right)^2}{2\frac{\widetilde{\sigma_e^2}}{X_j'X_j+\widetilde{\lambda}}}\right]$.

3.3.3 Data analyses

Real genotypic data and simulated phenotypic data were used here to compare these five methods. The genotypic data had 3961 individuals with 55,734 SNP markers. The heritability of the simulated trait was 0.25. The training data contained 3206 individuals and the remaining individuals were used for testing. A chain of length of 50,000 was used to estimate parameters of interest. Prediction accuracies were calculated using different samplers. The effective sample sizes (Geyer, 1992) were calculated for σ_e^2 to compare convergence rates for different methods. Computing time for different methods with the same number of iterations were also compared.

3.4 Results

The number of effective samples per second of computing time were obtained for BayesB using MH, efficient MH or the three different Gibbs samplers. These three Gibbs samplers were almost twice as efficient as Metropolis-Hastings (Table 7.5). The prediction accuracies based on posterior means of marker effects for different samplers are all 0.296. Posterior means of μ for these four samplers are all 2.508. Posterior means of σ_e^2 for different samplers are almost equal, ranging from 0.955 to 0.957.

3.5 Discussion

In the joint Gibbs sampler, δ_j and β_j are sampled jointly, which addresses the problem of dependence between δ_j and β_j. Thus, the joint sampler had the largest effective sample size. On the other hand, in the single-site Gibbs sampler, δ_j and β_j are sampled from their full conditionals, and thus due to the dependence between β_j and δ_j, the single-site Gibbs sampler had the smallest effective sample size. These differences in effective sample size, however, were negligible.

In the Gibbs sampler with pseudo priors, β_j and δ_j are also sampled from their full conditionals. Recall that we have shown the posterior mean of the marker effects does not depend on the pseudo prior. Furthermore, Godsill (Godsill, 2001) has shown that the marginal posterior for parameters in the model do not depend on the pseudo prior, which is the prior for β_j when

$\delta_j = 0$. As suggested by Carlin and Chib (Carlin and Chib, 1995), when the full conditional distribution of β_j when $\delta_j = 1$ is chosen to be the pseudo prior, we have shown that the samples of β_j and δ_j are identically distributed to those from the joint Gibbs sampler. Thus, the Gibbs sampler with pseudo priors will have similar effective sample size as the joint Gibbs sampler.

However, when the full conditional distribution for β_j when $\delta_j = 1$ is used as the pseudo prior in BayesB, the full conditional distributions of σ_e^2 and σ_j^2 are not of known forms because σ_e^2 and σ_j^2 are in the pseudo prior for the marker effect. In contrast to BayesB, in the model used by Godsill (Godsill, 2001) to justify the use of full conditional distributions as the pseudo priors, for simplicity, hyper-parameters such as σ_e^2 were omitted (Godsill, 2001). Here, we have replaced σ_e^2 and σ_j^2 in the pseudo prior with constants such that the full conditionals for σ_e^2 and σ_j^2 have scaled inverted chi-square distributions. This modification will give a pseudo prior whose distribution is close to that of the full conditional. In the Gibbs sampler with this pseudo prior, the effective sample size was smaller than in the joint Gibbs sampler but still larger than in the single-site Gibbs sampler.

When a MH algorithm is used to jointly sample the marker effect and the locus-specific variance, the BayesB method is computationally intensive. After introducing a variable δ_j, indicating whether the marker effect for a locus is zero or non-zero, the marker effect and locus-specific variance can be sampled using Gibbs sampler without MH. Among the Gibbs samplers that were considerd here, the joint Gibbs sampler is the most efficient. This sampler is about 2.1 times as efficient as the MH algorithm proposed by Meuwissen et al. (Meuwissen et al., 2001a) and 1.7 times as efficient as that proposed by Habier et al. (Habier et al., 2011a).

Author's contributions

HC, LQ, DG, RF contributed to the development of the statistical methods. HC wrote the program code and conducted the analyses. The manuscript was prepared by RF and HC. All authors read and approved the final manuscript.

Table 3.1 Efficiency of alternative MCMC samplers for BayesB. Results are given for the computing time in seconds to obtain 50,000 samples, effective sample size and effective samples/second for BayesB using Metropolis-Hastings (MH), single-site Gibbs sampler, joint Gibbs sampler and Gibbs sampler with pseudo priors.

	Alternative MCMC Samplers				
	MH	efficient MH	single-site Gibbs	joint Gibbs	Gibbs with pseudo prior
Computing Time	90,009	70,714	52,452	44,726	47,043
Effective Sample Size	25,262	24,588	24,684	26,757	25,036
Effective Samples/Second	0.280	0.347	0.471	0.598	0.532

CHAPTER 4. PARALLEL COMPUTING TO SPEED UP WHOLE-GENOME BAYESIAN REGRESSION ANALYSES USING ORTHOGONAL DATA AUGMENTATION

Hao Cheng, Dorian Garrick and Rohan Fernando

A paper to be submitted

4.1 Abstract

Bayesian multiple regression methods are widely used in whole-genome analyses to solve the problem that the number p of marker covariates is usually larger than the number n of observations. Inferences from most Bayesian methods are based on Markov chain Monte Carlo methods, where statistics are computed from a Markov chain constructed to have a stationary distribution equal to the posterior distribution of the unknown parameters. In practice, chains of about fifty thousand steps are typically used in whole-genome Bayesian regression analyses, which is computationally intensive. In this paper, we have shown how the sampling of marker effects can be made independent within each step of the chain. This is done by augmenting the marker covariate matrix by adding p new rows to it such that columns of the augmented marker covariate matrix are orthogonal. The phenotypes corresponding to the augmented rows of marker covariate matrix are considered missing. Ideally, the computations at each step of the MCMC chain, can be speeded up by the number k of computer processors up to the number p of markers. Addressing the heavy computational burden associated with Bayesian methods by parallel computing will lead to greater use of these methods.

4.2 Introduction

Genome-wide single nucleotide polymorphism (SNP) marker data have been adopted for whole genome analyses, including genomic prediction (Meuwissen et al., 2001a) and genome-wide association studies (Visscher et al., 2007). In whole-genome analyses, the number p of marker covariates is usually larger than the number n of observations. Bayesian multiple regression methods are widely used to address this problem, where the effects of all markers are estimated simultaneously combining the information from the phenotypic data and priors for the marker effects. Most widely-used Bayesian regression methods only differ in the prior used for the marker effects. For example, the prior for each marker effect in BayesA (Meuwissen et al., 2001a) follows a scaled t distribution, whereas several other Bayesian regression methods accommodate models where the prior for each marker effect follows a mixture distribution, such as BayesB (Meuwissen et al., 2001a), BayesC (Habier et al., 2011a) and BayesR (Erbe et al., 2012; Moser et al., 2015).

In these Bayesian regression analyses, closed-form expressions for the posterior distribution of parameters of interest, e.g., marker effects, are usually not available. Thus inferences from most Bayesian methods are based on Markov chain Monte Carlo (MCMC) methods, where statistics are computed from a Markov chain constructed to have a stationary distribution equal to the posterior distribution of the unknown parameters. Suppose $\mathbf{x}$ is a stochastic vector of unknown parameters of interest. A Markov chain $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3, \ldots$ is a sequence of $\mathbf{x}$, where the distribution of $\mathbf{x}_t$ at step t conditional on all the previous steps only depends on the distribution of $\mathbf{x}_{t-1}$ at step $t-1$. It has been shown that statistics computed from such a Markov chain converge to those from the stationary distribution as the chain length increases (Norris, 1997). In practice, chains of about fifty thousand steps are typically used in whole-genome Bayesian regression analyses (Fernando et al., 2016). Note that the vector $\mathbf{x}$ has length p or a multiple of it if auxiliary variables such as marker effect variances are introduced to the analysis as in BayesA or BayesB.

A widely used method to construct such a Markov chain is Gibbs sampling. In Gibbs sampling, at step t, each component of the vector $\mathbf{x}_t$ is sampled from the conditional distribution

of that component given all the other components sampled up to that point (Sorensen and Gianola, 2002). In a fast and efficient Gibbs sampler proposed for BayesB (Cheng et al., 2015b), for example, within each step, each variable in the vector $\mathbf{x}$ is sampled conditional on all the other variables. This includes, for each marker i, its effect, the effect variance and a Bernoulli variable indicating whether the effect is zero or non-zero, as well as the intercept and the residual variance. This is an example of a single-site Gibbs sampler where each variable is sampled at one time conditional on the current values of all other variables. In summary, whole-genome Bayesian multiple regression analyses require constructing Markov chains of length about fifty thousand. Within each step of the chain, Gibbs sampling requires sampling at least p unknowns. This makes Bayesian multiple regression analyses computationally intensive.

Parallel computing has been proposed to address this problem (Wu et al., 2012). Parallel computing refers to the use of multiple processors to perform computations in parallel. It is often suggested that a large number of shorter chains can be constructed in parallel and combine the statistics computed from these chains. However, the Ergodic theorem of Markov chain theory states that statistics computed from an increasingly long chain, rather than an increasing number of short chains, converge to those from the stationary distribution (Norris, 1997). Thus, combining several chains will reduce the Monte Carlo variance of the computed quantities, but this may not yield statistics from the stationary distribution. The problem with this approach is that a Markov chain is a sequential process, and thus it can not broken into several independent processes. However, a valid approach is to use Independent Metropolis-Hastings (IMH) sampling (Sorensen and Gianola, 2002), where a large number of candidate samples $\mathbf{x}_t$ are obtained independently using parallel computing. Then these candidate samples are accepted or rejected sequentially using the Metropolis-Hastings algorithm to construct a single long chain (Jacob et al., 2012).

Another approach is to parallelize the Gibbs sampling for each marker within each step of the chain. In single-site Gibbs sampler, however, sampling of each variable is from the full conditional distribution, which is conditional distribution of the variable given the current values of all other variables. Thus, parallel Gibbs sampling would not be feasible unless the full conditional distributions do not depend on the values of the variables being conditioned

on, i.e., unless the full-conditionals are independent. In this paper, we will show how the full conditional distributions of the marker effects can be made independent within each step of the chain. This is done by augmenting the marker covariate matrix by adding p new rows to it such that columns of the augmented marker covariate matrix are orthogonal (Ghosh, Joyee and Clyde, Merlise A, 2012). The phenotypes corresponding to the augmented rows of marker covariate matrix are considered missing (Ghosh, Joyee and Clyde, Merlise A, 2012).

The computations for obtaining samples of the marker effects involves vector additions and dot products of length n. Parallel computing can also be used to speed up these computations, where vectors are split up and additions or products are done in parallel on multiple processors (Fernando et al., 2014). This approach can be used within each parallel Gibbs sampling.

The objective of this paper is to show how the parallel Gibbs sampling approach using an augmented marker covariate matrix can be used in Bayesian multiple regression methods with the BayesC prior. Use of this approach with other priors, such as those in BayesA, BayesB or Bayesian Lasso, should be straightforward.

4.3 Methods

4.3.1 Model

In Bayesian regression, phenotypes of are often modeled as

$$\mathbf{y} = \mathbf{1}\mu + \mathbf{X}\boldsymbol{\alpha} + \mathbf{e},$$

where $\mathbf{y}$ is the vector of n phenotypes, μ is the overall mean, $\mathbf{X}$ is the $n \times p$ marker covariate matrix (coded as 0, 1, 2), $\boldsymbol{\alpha}$ is a vector of p random marker effects and $\mathbf{e}$ is a vector of n random residuals. A flat prior is used for μ. The prior for the residual $\mathbf{e}$ is $\mathbf{e}|\sigma_e^2 \sim N(0, \mathbf{I}\sigma_e^2)$ with $\left(\sigma_e^2 \mid \nu_e, S_e^2\right) \sim \nu_e S_e^2 \chi_{\nu_e}^{-2}$. The columns of $\mathbf{X}$ are usually centered. In BayesC, the prior for the marker effect is a mixture of a point mass at zero and a univariate normal distribution with null mean and a common locus variance σ_α^2 with $\left(\sigma_\alpha^2 \mid \nu_\alpha, S_\alpha^2\right) \sim \nu_\alpha S_\alpha^2 \chi_{\nu_\alpha}^{-2}$ (Habier et al., 2011a).

4.3.2 Parallel computing strategy using orthogonal data augmentation

4.3.2.1 Gibbs sampling for marker effects in BayesC

In Gibbs sampling for BayesC, the full conditional distribution of α_j, the marker effect for locus j, when α_j is non-zero, can be written as

$$(\alpha_j \mid ELSE) \sim N\left(\hat{\alpha_j}, \frac{\sigma_e^2}{\mathbf{X}_j^T\mathbf{X}_j + \frac{\sigma_e^2}{\sigma_\alpha^2}}\right),$$

where $ELSE$ stands for all the other unknowns and $\mathbf{y}$, $\mathbf{X}_j$ is the jth column of $\mathbf{X}$, and $\hat{\alpha_j}$ is the solution to

$$\begin{aligned}\left(\mathbf{X}_j^T\mathbf{X}_j + \frac{\sigma_e^2}{\sigma_\alpha^2}\right)\hat{\alpha_j} &= \mathbf{X}_j^T\left(\mathbf{y} - \mathbf{1}\mu - \sum_{j'\neq j}\mathbf{X}_{j'}\alpha_{j'}\right) \\ &= \mathbf{X}_j^T\mathbf{y} - \mathbf{X}_j^T\mathbf{1}\mu - \sum_{j'\neq j}\mathbf{X}_j^T\mathbf{X}_{j'}\alpha_{j'}. \qquad (4.1)\end{aligned}$$

In the Gibbs sampling, the sample for each marker, α_j, can not be obtained simultaneously in parallel, because samples for other marker effects, $\alpha_{j'\neq j}$, appear in the term $\sum_{j'\neq j}\mathbf{X}_j^T\mathbf{X}_{j'}\alpha_{j'}$ on the right-hand-side of (4.1), i.e., the full conditional distributions of the marker effects are not independent. One solution is to orthogonalize columns of the marker covariate matrix $\mathbf{X}$ such that the term $\sum_{j'\neq j}\mathbf{X}_j^T\mathbf{X}_{j'}\alpha_{j'}$ in (4.1) becomes zero. The data augmentation approach that is described below was proposed by Ghosh et al. (Ghosh, Joyee and Clyde, Merlise A, 2012) to obtain a design matrix with orthogonal columns.

4.3.2.2 Orthogonal Data Augmentation (ODA)

Let $\mathbf{W}_o = \begin{bmatrix}\mathbf{1} & \mathbf{X}\end{bmatrix}$ be the design matrix for the BayesC analysis. Following Ghosh et al. (Ghosh, Joyee and Clyde, Merlise A, 2012), we show here how to augment $\mathbf{W}_o$ as $\mathbf{W}_c = \begin{bmatrix}\mathbf{W}_o \\ \mathbf{W}_a\end{bmatrix}$ such that

$$\mathbf{W}_c^T\mathbf{W}_c = \begin{bmatrix}\mathbf{W}_o^T & \mathbf{W}_a^T\end{bmatrix}\begin{bmatrix}\mathbf{W}_o \\ \mathbf{W}_a\end{bmatrix} = \mathbf{D},$$

where $\mathbf{W}_a$ is a square matrix of dimension p and $\mathbf{D}$ is a diagonal matrix. Thus,

$$\mathbf{W}_a^T\mathbf{W}_a = \mathbf{D} - \mathbf{W}_o^T\mathbf{W}_o, \tag{4.2}$$

and $\mathbf{W}_a$ can be obtained using Cholesky decomposition (or Eigen decomposition) from (4.2). The choice of $\mathbf{D}$ is $\mathbf{I}d$, where d is set to be the largest eigenvalue of $\mathbf{W}_o^T\mathbf{W}_o$ (Ghosh, Joyee and Clyde, Merlise A, 2012). In practice, a small value, e.g., 0.001, was added to d to avoid computationally unstable solutions (Ghosh, Joyee and Clyde, Merlise A, 2012).

4.3.2.3 BayesC model with ODA (BayesC-ODA)

Employing ODA, the Bayesian regression model can be written as

$$\begin{bmatrix}\mathbf{y}\\ \widetilde{\mathbf{y}}\end{bmatrix} = \begin{bmatrix}\mathbf{1}\\ \widetilde{\mathbf{J}}\end{bmatrix}\mu + \begin{bmatrix}\mathbf{X}\\ \widetilde{\mathbf{X}}\end{bmatrix}\boldsymbol{\alpha} + \begin{bmatrix}\mathbf{e}\\ \widetilde{\mathbf{e}}\end{bmatrix}, \tag{4.3}$$

where $\widetilde{\mathbf{y}}$ denotes a vector of unobserved phenotypes that are introduced into the model, $\begin{bmatrix}\mathbf{e}\\ \widetilde{\mathbf{e}}\end{bmatrix} \sim N\left(\mathbf{0}, \mathbf{I}\sigma_e^2\right)$ and $\widetilde{\mathbf{J}}$, $\widetilde{\mathbf{X}}$ are obtained using (4.2) with $\mathbf{W}_a = \begin{bmatrix}\widetilde{\mathbf{J}} & \widetilde{\mathbf{X}}\end{bmatrix}$,$\mathbf{W}_o = \begin{bmatrix}\mathbf{1} & \mathbf{X}\end{bmatrix}$.

In BayesC-ODA, the full conditional distribution of $\boldsymbol{\alpha}$ under model (4.3), which was derived in the Appendix, can be written as

$$(\alpha_j \mid ELSE) \sim N\left(\frac{\mathbf{X}_j^T\mathbf{y} + \widetilde{\mathbf{X}}_j^T\widetilde{\mathbf{y}}}{d + \frac{\sigma_e^2}{\sigma_\alpha^2}}, \frac{\sigma_e^2}{d + \frac{\sigma_e^2}{\sigma_\alpha^2}}\right), \tag{4.4}$$

where the mean and variance parameters are free of the values of the other marker effects $\alpha_{j'\neq j}$. Thus the full conditional distribution of the marker effects are independent, and thus, samples for each marker can be obtained simultaneously in parallel. At each step of the MCMC chain, the "missing" phenotypes $\widetilde{\mathbf{y}}$ are sampled from

$$(\widetilde{\mathbf{y}} \mid ELSE) \sim N\left(\widetilde{\mathbf{J}}\mu + \widetilde{\mathbf{X}}\alpha, \mathbf{I}\sigma_e^2\right). \tag{4.5}$$

The derivation of the full conditional distributions of other parameters of interest are shown in the Appendix.

4.3.2.4 Simulated data

Simulated genotypic and phenotypic data were used to compare BayesC and BayesC-ODA. The simulated genome consisted of 10 chromosomes each 5 cM long and containing 50 evenly spaced loci. Allele states were sampled from a Bernoulli distribution with frequency 0.5. A random sample of 25 loci were selected as QTL, and their effects were sampled from a univariate normal distribution with mean zero and variance one. Starting from a base population of 100 males and 100 females, random mating was simulated for 100 generations to generate linkage disequilibrium. In generation 101, the population size was increased to 3000 males and 3000 females, and random mating was continued for four more generations. The QTL effects were scaled such that the genetic variance for a randomly sampled individual from generation 105 was 1.0. Phenotypes were simulated by adding independent residuals that were sampled from a normal distribution with null mean and variance one to the genetic values. To investigate the performance of BayesC-ODA with $n < p$ or $n > p$, 100 or 5000 individuals were used for training. A population of 1000 individuals was used for testing. In the testing population, estimated breeding values were calculated using BayesC and BayesC-ODA. Correlation between estimated breeding values or estimated marker effects from BayesC-ODA and BayesC was investigated for a chain of length 5,000,000 to study: 1) whether BayesC-ODA provided identical estimated marker effects and breeding values as BayesC; 2) the convergence of BayesC-ODA.

The true genetic variance and residual variance were used to calculate the scale parameters of the inverse-chisquare priors of the residual variance and marker effect variance (Fernando and Garrick, 2013a).

4.4 Results

The correlation between estimated breeding values for the testing population from BayesC and BayesC-ODA by chain length was investigated. In the scenario where $n < p$, this correlation was larger than 0.99 when the chain was longer than 9,000 and became larger than 0.999 as the chain grew longer than 75,000. In the scenario where $n > p$, this correlation was larger than

0.99 when the chain was longer than 1,000 and became 0.999 as the chain grew longer than 18,000.

The correlation between posterior mean of marker effects from BayesC and BayesC-ODA as the chain length increases was investigated. In the scenario where $n < p$, this correlation was larger than 0.99 when the chain was longer than 37,000 and became larger than 0.999 as the chain grew longer than 439,000. In the scenario where $n > p$, this correlation was larger than 0.99 when the chain was longer than 649,000 and became about 0.999 as the chain length reached 5,000,000.

4.5 Discussion

Whole-genome Bayesian multiple regression methods are usually computationally intensive, where a MCMC chain of about fifty thousand steps is typically used for inference. In this paper, a strategy to parallelize Gibbs sampling for each marker within each step of the MCMC chain was proposed. This parallelization is accomplished by using an orthogonal data augmentation strategy, where the marker covariate matrix is augmented by adding p new rows such that its columns are orthogonal (Ghosh, Joyee and Clyde, Merlise A, 2012). Then, the full conditional distributions of marker effects become independent within each step of the chain, and thus, samples of marker effects within each step can be drawn in parallel. In this paper, the full conditional distributions that are needed for BayesC with orthogonal data augmentation (BayesC-ODA) were derived and the convergence of BayesC-ODA was studied. In analyses of the simulated data, BayesC-ODA provided virtually identical predictions of breeding values as BayesC when the chain length was about 20,000 to 80,000, which is similar to the commonly used chain length of 50,000. Some ideas for parallel implementation of BayesC-ODA are briefly discussed below with more details in the appendix. The investigation of these ideas and parallel implementation of Bayesian multiple regression with ODA will be undertaken in a separate study.

In Bayesian multiple regression methods such as BayesC, the most time consuming task is sampling the marker effects from their full conditional distributions. In BayesC-ODA, however, the marker effects within each step can be sampled in parallel, using (4.4). Ideally, the compu-

tations at each step of the MCMC chain, can be speeded up by the number k of processors up to the number p of markers. However, two extra computations are required in BayesC-ODA. The first is sampling of the vector $\widetilde{\mathbf{y}}$ of unobserved phenotypes, which is required in each MCMC step. Each element of $\widetilde{\mathbf{y}}$ is sampled from an independent univariate normal distribution with the variance equal to the current value of σ_e^2. The means of these normal distributions can be computed in parallel as described in the appendix. Once the means are computed, each element in $\widetilde{\mathbf{y}}$ can be sampled in parallel. The second is the computation of the augmented matrix $\mathbf{W}_a$ as in (4.2), which is required only once at the beginning of the MCMC chain. In (4.2), there are two computationally intensive tasks: 1) computation of $\mathbf{X}^T\mathbf{X}$, where $\mathbf{X}$ is a $n \times p$ matrix; and 2) Cholesky decomposition of a positive definite matrix of size p. Parallel computing approaches for the first of these two tasks is given in the appendix. The computing time for the Cholesky decomposition in the second task is relatively short, taking only a few minutes for $p = 50,000$ on a workstation, using one graphics processing unit (GPU).

It is worth noting that two approaches are available to compute the right-hand-side of (4.1). In the first approach, equation (29) in (Fernando et al., 2014) is used, where number of operations is of order n. In the second approach, equation (33) in (Fernando et al., 2014) is used, where the number of operations is of order p. In BayesC-ODA, the first approach is used. As can be seen from (4.1), the right-hand-side for α_j is $\mathbf{X}_j^T\mathbf{y} + \widetilde{\mathbf{X}}_j^T\widetilde{\mathbf{y}}$, where $\mathbf{X}_j^T\mathbf{y}$ is constant, and only $\widetilde{\mathbf{X}}_j^T\widetilde{\mathbf{y}}$ needs to be computed at each step of the MCMC chain, where the number of operations for this is always of order p regardless of the size of n. However, when the first approach is used for multiple-trait BayesC analyses, the size of the dataset that can be analyzed is limited by the requirement to store the entire marker covariate matrix of size $n \times p$ in memory so that $\mathbf{y} - \mathbf{1}\mu - \sum \mathbf{X}_j\alpha_j$ can be updated with the current value of α_j. So, as n grows, this approach will become infeasible. On the other hand, in BayesC-ODA, only $\widetilde{\mathbf{X}}$ of constant size $p \times p$ needs to be stored in memory regardless of the size of n, which is required in (4.4) and (4.5). Thus, even when n grows, multiple-trait analyses will only require storing a $p \times p$ matrix regardless of the number of traits and n.

We have shown here that the predictions of breeding values from BayesC-ODA converge to those from BayesC but may require a chain of 80,000 steps as opposed to one of 50,000 for

BayesC. However, Gibbs sampling of marker effects within each step can be done in parallel for BayesC-ODA, and this is expected to result in a considerable speedup for BayesC-ODA. Further, as discussed above, multiple-trait analyses with BayesC-ODA only require storing the p augmented rows of the covariate matrix regardless of the number of traits and observations. Thus, when n is large, BayesC-ODA may provide an efficient approach for multiple-trait Bayesian regression analyses.

Author's contributions

HC, RF contributed to the development of the statistical methods. HC wrote the program code and conducted the analyses. The manuscript was prepared by HC and RF. All authors read and approved the final manuscript.

4.6 Appendix

In many modern programming languages, such as R, Python and Julia, libraries are available to take advantage of multiple processors and GPUs for parallel computing of many matrix or vector operations. The descriptions given below are only to illustrate the main principle underlying parallel computing of splitting up calculations across processors. Actual implementations may be different and will depend on the programming language, the library and the hardware used.

4.6.1 Parallel Computing of Ab

To sample the unobserved phenotypic values using (4.5), a matrix by vector product $\widetilde{\mathbf{X}}\boldsymbol{\alpha}$ is needed. Here we describe how parallel computing can be used to compute the product of a matrix $\mathbf{A}$ by a vector $\mathbf{b}$.

1. Split $\mathbf{A}$ of size $n \times p$ by columns into smaller matrices $\mathbf{A}_{(1)}, \mathbf{A}_{(2)}, \mathbf{A}_{(3)}, \ldots$of size $n \times p_i$, and split $\boldsymbol{\alpha}$ into smaller vectors $\mathbf{b}_{(1)}, \mathbf{b}_{(2)}, \mathbf{b}_{(3)}, \ldots$ of length p_i with $\sum p_i = p$.

2. Compute $\mathbf{Ab}$ as $\mathbf{A}_{(1)}\mathbf{b}_{(1)} + \mathbf{A}_{j(2)}\mathbf{b}_{(2)} + \mathbf{A}_{j(3)}\mathbf{b}_{(3)} + \ldots$, where $\mathbf{A}_{(i)}\mathbf{b}_{(i)}$ for $i = 1, 2, \ldots$ are computed on different processors and then summed to obtain $\mathbf{Ab}$.

The same strategy can also be used to calculate $\mathbf{X}^T\mathbf{y}$ by splitting $\mathbf{X}$ by rows.

4.6.2 Parallel Computing of $\mathbf{A}^T\mathbf{A}$

In (4.2), computation of $\mathbf{X}^T\mathbf{X}$ is needed. Here we describe how parallel computing can be used to compute $\mathbf{A}^T\mathbf{A}$, where $\mathbf{A}$ is a $n \times p$ matrix.

1. Split $\mathbf{X}$ of size $n \times p$ by rows into smaller matrices $\mathbf{A}_{(1)}, \mathbf{A}_{(2)}, \mathbf{A}_{(3)}, \ldots$of size $n_i \times p$ with $\sum n_i = n$.

2. Compute $\mathbf{A}^T\mathbf{A} = \sum_{j=1}^{k} \mathbf{A}_{(j)}^T\mathbf{A}_{(j)}$, where $\mathbf{A}_{(j)}^T\mathbf{A}_{(j)}$ for $j = 1, 2, \ldots$ are computed on different processors and then summed to obtain $\mathbf{A}^T\mathbf{A}$.

In addition to reducing the computing time, this approach can also address the limitation that $\mathbf{A}$ may be too large to be stored on a single computing node by distributing the $\mathbf{A}_{(i)}$ across several nodes.

4.6.3 Single-site Gibbs sampler for BayesC-ODA

4.6.3.1 full conditional distribution of the marker effect

Detailed derivation of the full conditional distributions of the marker effect for locus j in BayesC is in Fernando and Garrick (Fernando and Garrick, 2013a). As shown in (Fernando and Garrick, 2013a), the full conditional distribution of α_j in BayesC, when α_j is non-zero, is

$$(\alpha_j \mid ELSE) \sim N\left(\hat{\alpha_j}, \frac{\sigma_e^2}{\mathbf{X}_j^T\mathbf{X}_j + \frac{\sigma_e^2}{\sigma_\alpha^2}}\right), \tag{4.6}$$

where $ELSE$ stands for all the other unknowns and $\mathbf{y}$, $\mathbf{X}_j$ is the jth column of $\mathbf{X}$, and $\hat{\alpha_j}$ is the solution to

$$\left(\mathbf{X}_j^T\mathbf{X}_j + \frac{\sigma_e^2}{\sigma_\alpha^2}\right)\hat{\alpha_j} = \mathbf{X}_j^T\left(\mathbf{y} - \mathbf{1}\mu - \sum_{j' \neq j} \mathbf{X}_{j'}\alpha_{j'}\right). \tag{4.7}$$

The full conditional distribution of α_j in BayesC-ODA, which is shown below, can be obtained from (4.6) and (4.7) by replacing $\mathbf{y}$ with $\begin{bmatrix}\mathbf{y}\\ \widetilde{\mathbf{y}}\end{bmatrix}$, $\mathbf{1}$ with $\begin{bmatrix}\mathbf{1}\\ \widetilde{\mathbf{J}}\end{bmatrix}$ and $\mathbf{X}$ with $\begin{bmatrix}\mathbf{X}\\ \widetilde{\mathbf{X}}\end{bmatrix}$. Note that columns

of the augmented covariate matrix $\begin{bmatrix} \mathbf{1} & \mathbf{X} \\ \widetilde{\mathbf{J}} & \widetilde{\mathbf{X}} \end{bmatrix}$ are orthogonal. Thus, (4.7) for BayesC-ODA can be simplified as

$$\left(\begin{bmatrix}\mathbf{X}_j^T\widetilde{\mathbf{X}}_j^T\end{bmatrix}\begin{bmatrix}\mathbf{X}_j\\ \widetilde{\mathbf{X}}_j\end{bmatrix}+\frac{\sigma_e^2}{\sigma_\alpha^2}\right)\hat{\alpha}_j=\begin{bmatrix}\mathbf{X}_j^T\widetilde{\mathbf{X}}_j^T\end{bmatrix}\left(\begin{bmatrix}\mathbf{y}\\ \widetilde{\mathbf{y}}\end{bmatrix}-\begin{bmatrix}\mathbf{1}\\ \widetilde{\mathbf{J}}\end{bmatrix}\mu-\sum_{j'\neq j}\begin{bmatrix}\mathbf{X}_{j'}\\ \widetilde{\mathbf{X}}_{j'}\end{bmatrix}\alpha_{j'}\right)$$

$$\left(\begin{bmatrix}\mathbf{X}_j^T\widetilde{\mathbf{X}}_j^T\end{bmatrix}\begin{bmatrix}\mathbf{X}_j\\ \widetilde{\mathbf{X}}_j\end{bmatrix}+\frac{\sigma_e^2}{\sigma_\alpha^2}\right)\hat{\alpha}_j=\begin{bmatrix}\mathbf{X}_j^T\widetilde{\mathbf{X}}_j^T\end{bmatrix}\begin{bmatrix}\mathbf{y}\\ \widetilde{\mathbf{y}}\end{bmatrix}-\begin{bmatrix}\mathbf{X}_j^T\widetilde{\mathbf{X}}_j^T\end{bmatrix}\begin{bmatrix}\mathbf{1}\\ \widetilde{\mathbf{J}}\end{bmatrix}\mu-\sum_{j'\neq j}\begin{bmatrix}\mathbf{X}_j^T\widetilde{\mathbf{X}}_j^T\end{bmatrix}\begin{bmatrix}\mathbf{X}_{j'}\\ \widetilde{\mathbf{X}}_{j'}\end{bmatrix}\alpha_{j'}$$

$$\left(d+\frac{\sigma_e^2}{\sigma_\alpha^2}\right)\hat{\alpha}_j=\begin{bmatrix}\mathbf{X}_j^T\widetilde{\mathbf{X}}_j^T\end{bmatrix}\begin{bmatrix}\mathbf{y}\\ \widetilde{\mathbf{y}}\end{bmatrix}$$

$$\hat{\alpha}_j=\frac{\mathbf{X}_j^T\mathbf{y}+\widetilde{\mathbf{X}}_j^T\widetilde{\mathbf{y}}}{d+\frac{\sigma_e^2}{\sigma_\alpha^2}}. \tag{4.8}$$

Thus, the full conditional distribution of α_j can be written as

$$(\alpha_j \mid ELSE) \sim N\left(\frac{\mathbf{X}_j^T\mathbf{y}+\widetilde{\mathbf{X}}_j^T\widetilde{\mathbf{y}}}{d+\frac{\sigma_e^2}{\sigma_\alpha^2}}, \frac{\sigma_e^2}{d+\frac{\sigma_e^2}{\sigma_\alpha^2}}\right).$$

Detailed derivation of the full conditional distribution of the indicator variable δ_j indicating if α_j had a normal distribution ($\delta_j = 1$) or if it is null ($\delta_j = 0$) in BayesC is also in Fernando and Garrick (Fernando and Garrick, 2013a). The full conditional distribution of δ_j in BayesC is

$$Pr\left(\delta_j = 1 \mid ELSE\right) = \frac{f_1\left(r_j \mid \sigma_\alpha^2, \sigma_e^2\right) Pr\left(\delta_j = 1\right)}{f_0\left(r_j \mid \sigma_e^2\right) Pr\left(\delta_j = 0\right) + f_1\left(r_j \mid \sigma_\alpha^2, \sigma_e^2\right) Pr\left(\delta_j = 1\right)}, \tag{4.9}$$

where $f_1\left(r_j \mid \sigma_\alpha^2, \sigma_e^2\right)$ is a univariate normal with

$$E\left(r_i \mid \sigma_\alpha^2, \sigma_e^2\right) = 0, Var\left(r_i \mid \sigma_\alpha^2, \sigma_e^2\right) = \left(\mathbf{X}_j^T\mathbf{X}_j\right)^2\sigma_\alpha^2 + \mathbf{X}_j^T\mathbf{X}_j\sigma_e^2,$$

and $f_0\left(r_j \mid \sigma_\alpha^2, \sigma_e^2\right)$ is a univariate normal with

$$E\left(r_i \mid \sigma_e^2\right) = 0, Var\left(r_i \mid \sigma_e^2\right) = \mathbf{X}_j^T\mathbf{X}_j\sigma_e^2,$$

and

$$r_j = \mathbf{X}_j^T\left(\mathbf{y} - \mathbf{1}\mu - \sum_{j'\neq j}\mathbf{X}_{j'}\alpha_{j'}\right).$$

The full conditional distribution of δ_j in BayesC-ODA, which is shown below, can be obtained from (4.9) by replacing $\mathbf{y}$ with $\begin{bmatrix}\mathbf{y}\\ \widetilde{\mathbf{y}}\end{bmatrix}$, $\mathbf{1}$ with $\begin{bmatrix}\mathbf{1}\\ \widetilde{\mathbf{J}}\end{bmatrix}$ and $\mathbf{X}$ with $\begin{bmatrix}\mathbf{X}\\ \widetilde{\mathbf{X}}\end{bmatrix}$. Thus, (4.9) for BayesC-ODA can be simplified as

$$Pr\left(\delta_j = 1 \mid ELSE\right) = \frac{f_1\left(r_j \mid \sigma^2_\alpha, \sigma^2_e\right) Pr\left(\delta_j = 1\right)}{f_0\left(r_j \mid \sigma^2_e\right) Pr\left(\delta_j = 0\right) + f_1\left(r_j \mid \sigma^2_\alpha, \sigma^2_e\right) Pr\left(\delta_j = 1\right)},$$

where $f_1\left(r_j \mid \sigma^2_\alpha, \sigma^2_e\right)$ is a univariate normal with

$$E\left(r_i \mid \sigma^2_\alpha, \sigma^2_e\right) = 0, Var\left(r_i \mid \sigma^2_\alpha, \sigma^2_e\right) = d^2\sigma^2_\alpha + d\sigma^2_e,$$

and $f_0\left(r_j \mid \sigma^2_e\right)$ is a univariate normal with

$$E\left(r_i \mid \sigma^2_e\right) = 0, Var\left(r_i \mid \sigma^2_e\right) = d\sigma^2_e,$$

and

$$r_j = \mathbf{X}_j^T\mathbf{y} + \widetilde{\mathbf{X}}_j^T\widetilde{\mathbf{y}}.$$

4.6.3.2 full conditional distributions of the unobserved phenotypes

The full conditional distribution of $\widetilde{\mathbf{y}}$ can be written as

$$\begin{aligned} f\left(\widetilde{\mathbf{y}} \mid \boldsymbol{\alpha}, \mu, \sigma^2_e, \sigma^2_\alpha, \mathbf{y}\right) &= f\left(\widetilde{\mathbf{y}} \mid \boldsymbol{\alpha}, \mu, \sigma^2_e\right) \\ &\propto N\left(\widetilde{\mathbf{J}}\mu + \widetilde{\mathbf{X}}\alpha, \mathbf{I}\sigma^2_e\right). \end{aligned}$$

4.6.3.3 full conditional distributions of other unknowns

The derivation of the full conditional distributions of other parameters such as μ, σ^2_α, σ^2_e are straightforward. Thus they are presented as below without derivations.

$$\left(\mu \mid ELSE\right) \sim N\left(\frac{\mathbf{1}^T\mathbf{y} + \widetilde{\mathbf{J}}^T\widetilde{\mathbf{y}}}{d}, \frac{\sigma^2_e}{d}\right);$$

$$\left(\sigma^2_\alpha \mid ELSE\right) \sim \left(\boldsymbol{\alpha}^T\boldsymbol{\alpha} + \nu_\alpha S^2_\alpha\right)\chi^{-2}_{k+\nu_\alpha};$$

$$\left(\sigma^2_e \mid ELSE\right) \sim \left(\boldsymbol{y}^T_{corr}\boldsymbol{y}_{corr} + \nu_\alpha S^2_\alpha\right)\chi^{-2}_{n+\nu_e},$$

where $\boldsymbol{y}_{corr} = \begin{bmatrix}\mathbf{y}\\ \widetilde{\mathbf{y}}\end{bmatrix} - \begin{bmatrix}\mathbf{1}\\ \widetilde{\mathbf{J}}\end{bmatrix}\mu - \sum\begin{bmatrix}\mathbf{X}_j\\ \widetilde{\mathbf{X}}_j\end{bmatrix}\alpha_j$ and k is the number of markers in the model.

CHAPTER 5. MULTIPLE-TRAIT BAYESIAN REGRESSION METHODS WITH MIXTURE PRIORS FOR GENOMIC PREDICTION

Hao Cheng, Kadir Kizilkaya, Jian Zeng, Dorian Garrick and Rohan Fernando

A paper submitted to Genetics

5.1 Abstract

Bayesian multiple-regression methods incorporating different mixture priors for marker effects are widely used in genomic prediction. Improvement in prediction accuracies from using those methods, such as BayesB, BayesC and BayesCπ, have been shown in single-trait analyses with both simulated and real data. These methods have been extended to multi-trait analyses, but only under a specific limited circumstance that assumes a locus affects all the traits or none of them. In this paper, we develop and implement the most general multi-trait BayesCΠ and BayesB methods allowing a broader range of mixture priors. Further, we compare them to single-trait methods and the "restricted" multi-trait formulation using real and simulated data. In those data analyses, significantly higher prediction accuracies were sometimes observed from these new broad-based multi-trait Bayesian multiple-regression methods. The software tool JWAS offers routines to perform the analyses.

5.2 Introduction

Genomic prediction was proposed by Meuwissen et al. (Meuwissen et al., 2001b) to incorporate marker effects from whole-genome data into genetic evaluation. In genomic prediction, all the marker or haplotype effects are estimated simultaneously, and these estimates can then

be used to predict breeding values of individuals not in the training population used to estimate the effects.

Bayesian multiple-regression methods incorporating mixture priors for marker effects are widely used in genomic prediction. For example, BayesB accommodates models where the prior for each marker effect follows a mixture distribution with a point mass at zero with probability π and a univariate-t distribution with probability $1-\pi$ (Meuwissen et al., 2001b; Cheng et al., 2015b). Another mixture model, BayesC, assumes a common locus variance for all marker effects, and its extension known as BayesCπ further treats π as an unknown parameter with a uniform prior distribution (Habier et al., 2011b).

Bayesian multiple-regression methods were first proposed for single-trait analyses but have been extended to some particular forms of multi-trait analyses (Calus and Veerkamp, 2011; Jia and Jannink, 2012). Those extensions have pertained to a particular, somewhat restrictive mixture model. The "restricted" multi-trait BayesCΠ presented by Jia et al. (Jia and Jannink, 2012) assumes a locus affects none of the traits or has simultaneous effects on all traits. This assumption of genetic architecture in that multi-trait BayesCΠ circumstance is violated if some loci have no effect on at least one of the traits while having an effect on at least one of the other traits.

In this paper, we present a more general class of multi-trait BayesCΠ and BayesB methods for which the previous restricted multi-trait model is a special case. The new methods are compared to single-trait methods and the previous multi-trait methods using real and simulated data.

5.3 Materials and Methods

5.3.1 Multi-trait Marker Effects Model

For simplicity and without loss of generality, we will assume individuals have all traits measured with a general mean as the only fixed effect, and write the multi-trait model for

individual i from n genotyped individuals as

$$\boldsymbol{y}_i = \boldsymbol{\mu} + \sum_{j=1}^{p} m_{ij}\boldsymbol{\alpha}_j + \boldsymbol{e}_i,$$

where $\boldsymbol{y}_i$ is a vector of phenotypes of t traits for individual i, $\boldsymbol{\mu}$ is a vector of overall means for t traits, m_{ij} is the genotype covariate at locus j for individual i (coded as 0,1,2), p is the number of genotyped loci, $\boldsymbol{\alpha}_j$ is a vector of allele substitution effects or marker effects of t traits for locus j, and $\boldsymbol{e}_i$ is a vector of random residuals of t traits for individual i. The fixed effects, or general mean in this case, are assigned flat priors. The residuals, $\boldsymbol{e}_i$, are *a priori* assumed to be independently and identically distributed multivariate normal vectors with null mean and covariance matrix $\boldsymbol{R}$, which in turn is assumed to have an inverse Wishart prior distribution, $W_t^{-1}(\mathbf{S}_e, \nu_e)$.

We will show that, employing the concept of data augmentation, the vector of marker effects at a particular locus $\boldsymbol{\alpha}_j$ can be written as $\boldsymbol{\alpha}_j = \boldsymbol{D}_j\boldsymbol{\beta}_j$, where $\boldsymbol{D}_j$ is a diagonal matrix whose kth diagonal entry is an indicator variable indicating whether the marker effect of locus j for trait k is zero or non-zero, and $\boldsymbol{\beta}_j$ follows a multivariate normal distribution in multi-trait BayesCΠ or a multivariate t distribution in multi-trait BayesB.

5.3.2 Multi-trait BayesCΠ model

5.3.2.1 Priors for marker effects

The prior for α_{jk}, the allele substitution or marker effect of trait k for locus j, is a mixture with a point mass at zero and a univariate normal distribution conditional on σ_k^2:

$$\alpha_{jk} \mid \pi_k, \sigma_k^2 \begin{cases} \sim N\left(0,\, \sigma_k^2\right) & probability\ (1-\pi_k) \\ 0 & probability\ \pi_k \end{cases}$$

and the covariance between effects for traits k and k' at the same locus, i.e., α_{jk} and $\alpha_{jk'}$ is

$$cov\left(\alpha_{jk}, \alpha_{jk'} \mid \sigma_{kk'}\right) = \begin{cases} \sigma_{kk'} & if\ both\ \alpha_{jk} \neq 0\ and\ \alpha_{jk'} \neq 0 \\ 0 & otherwise \end{cases}.$$

The vector of marker effects at a particular locus $\boldsymbol{\alpha}_j$ can be written as $\boldsymbol{\alpha}_j = \boldsymbol{D}_j\boldsymbol{\beta}_j$, where $\boldsymbol{D}_j$ is a diagonal matrix with elements $diag\,(\boldsymbol{D}_j) = \boldsymbol{\delta}_j = (\delta_{j1}, \delta_{j2}, \delta_{j3} \ldots \delta_{jt})$, where δ_{jk} is an indicator variable indicating whether the marker effect of locus j for trait k is zero or non-zero, and the vector $\boldsymbol{\beta}_j$ follows a multivariate normal distribution with null mean and covariance matrix $\boldsymbol{G} = \begin{bmatrix} \sigma_1^2 & \cdots & \sigma_{1t} \\ \vdots & \ddots & \vdots \\ \sigma_{1t} & \cdots & \sigma_t^2 \end{bmatrix}$. The covariance matrix $\boldsymbol{G}$ is *a priori* assumed to follow an inverse Wishart distribution, $W_t^{-1}(\mathbf{S}_\beta, \nu_\beta)$. Thus the multi-trait BayesCΠ model with data augmentation is written as

$$\boldsymbol{y}_i = \boldsymbol{\mu} + \sum_{j=1}^{p} m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j + \boldsymbol{e}_i. \tag{5.1}$$

In the most general case, any marker effect might be zero for any possible combination of t traits resulting in 2^t possible combinations of $\boldsymbol{\delta}_j$. For example, in a t=2 trait model, there are $2^t = 4$ combinations for $\boldsymbol{\delta}_j$, namely $\boldsymbol{\delta}_1 = (0, 0)$, $\boldsymbol{\delta}_2 = (0, 1)$, $\boldsymbol{\delta}_3 = (1, 0)$, $\boldsymbol{\delta}_4 = (1, 1)$. In the restricted special case of this model described by (Jia and Jannink, 2012), only $\boldsymbol{\delta}_1 = (0, 0)$ and $\boldsymbol{\delta}_4 = (1, 1)$ have non-zero probability. Suppose in general we use numerical labels "1", "2",..., "l" for the 2^t possible outcomes for $\boldsymbol{\delta}_j$, then the prior for $\boldsymbol{\delta}_j$ is a categorical distribution

$$\begin{aligned} & p\,(\boldsymbol{\delta}_j = \text{“}i\text{”}) \\ =& \Pi_1 I\,(\boldsymbol{\delta}_j = \text{“1”}) + \Pi_2 I\,(\boldsymbol{\delta}_j = \text{“2”}) + ... + \Pi_l I\,(\boldsymbol{\delta}_j = \text{“}l\text{”})\,, \end{aligned}$$

where $\sum_{i=1}^{l} \Pi_i = 1$ with Π_i being the prior probability that the vector $\boldsymbol{\delta}_j$ corresponds to the vector labelled "i".

A Dirichlet distribution with all parameters equal to one, i.e., a uniform distribution, can be used for the prior for $\boldsymbol{\Pi} = (\Pi_1, \Pi_2, ..., \Pi_l)$.

As shown below, we consider two Gibbs samplers to draw samples for all the parameters in this model. Gibbs sampler I requires that all 2^t outcomes for $\boldsymbol{\delta}_j$ have non-zero prior probabilities, i.e. none of Π_i can be zero. However, Gibbs sampler II does not. For example, in a 2 trait model, Gibbs sampler I requires that all 4 possible outcomes for $\boldsymbol{\delta}_j$ have non-zero probabilities, whereas Gibbs sampler II can accommodate models such as the one allowing only 2 outcomes

for $\boldsymbol{\delta}_j$: (0, 0),(1, 1). Gibbs sampler I would fail in that situation as the markov chain it generates is not irreducible, i.e. it is impossible to get to (0, 0) from (1, 1) or vice versa for $\boldsymbol{\delta}_j$ when single-site sampling is used and only one of the t indicator labels are sampled at a time.

5.3.2.2 Gibbs sampler I for multi-trait BayesCΠ

Suppose the prior for $\boldsymbol{\delta}_j$ is a categorical distribution whose support is for all 2^t possible outcomes of $\boldsymbol{\delta}_j$. For convenience, from now on let "1" denote trait k and "2" the other $t-1$ traits. In our sampling scheme, β_{j1} and δ_{j1} are sampled from their joint full conditional distributions, which can be written as the product of the full conditional distribution of β_{j1} given δ_{j1} and the marginal full conditional distribution of δ_{j1}. Let $\boldsymbol{\theta}$ denote all other parameters except δ_{j1} and β_{j1}, then our sampling scheme can be written as

$$f\left(\beta_{j1}, \delta_{j1} \mid \boldsymbol{\theta}, \boldsymbol{y}\right) = f\left(\beta_{j1} \mid \delta_{j1}, \boldsymbol{\theta}, \boldsymbol{y}\right) f\left(\delta_{j1} \mid \boldsymbol{\theta}, \boldsymbol{y}\right).$$

The full conditional distributions of β_{j1}, δ_{j1}, $\boldsymbol{\Pi}$, $\boldsymbol{G}$ and $\boldsymbol{R}$ for Gibbs sampler I, whose derivations are in the Appendix, are given below.

The full conditional distributions of β_{j1} is

$$p\left(\beta_{j1} \mid \delta_{j1}, \boldsymbol{\theta}, \boldsymbol{y}\right) = \begin{cases} N\left(\hat{\beta}^0_{j1}, \left(G^{11}\right)^{-1}\right) & when\ \delta_{j1} = 0 \\ N\left(\hat{\beta}^1_{j1}, \left(C^1_{j,11}\right)^{-1}\right) & when\ \delta_{j1} = 1 \end{cases},$$

with

$$\begin{aligned} \hat{\beta}^0_{j1} &= -\left(G^{11}\right)^{-1} \mathbf{G}^{12} \boldsymbol{\beta}_{j2}, \\ \hat{\beta}^1_{j1} &= \left(C^1_{j,11}\right)^{-1} \left(r_{j1} - \mathbf{C}^1_{j,12} \boldsymbol{\beta}_{j2}\right), \\ C^1_{j,11} &= G^{11} + R^{11} \sum_{i=1}^{n} m^2_{ij} \\ \mathbf{C}^1_{j,12} &= \boldsymbol{G}^{12} + \boldsymbol{R}^{12} \boldsymbol{D}_{j2} \sum_{i=1}^{n} m^2_{ij}, \\ r_{j1} &= \left(\sum_{i=1}^{n} \boldsymbol{w}_i' m_{ij}\right) \begin{bmatrix} R^{11} \\ \boldsymbol{R}^{21} \end{bmatrix}, \end{aligned}$$

where $\boldsymbol{w}_i = \boldsymbol{y}_i - \boldsymbol{\mu} - \sum_{j' \neq j} m_{ij'} \boldsymbol{D}_{j'} \boldsymbol{\beta}_{j'}$.

The marginal full conditional probability that $\delta_{j1} = 1$ is

$$f\left(\delta_{j1} = 1 \mid \boldsymbol{\theta}, \boldsymbol{y}\right) = \left\{1 + \left(\frac{Pr\left(\delta_{j1} = 0, \boldsymbol{\delta}_{j2} | \boldsymbol{\Pi}\right)}{Pr\left(\delta_{j1} = 1, \boldsymbol{\delta}_{j2} | \boldsymbol{\Pi}\right)} H\right)^{-1}\right\}^{-1},$$

where $H =$

$$exp\left\{-\frac{1}{2}\left(logC^1_{j,11} - \hat{\beta^1_{j1}}^{\,2} C^1_{j,11}\right) - \left(-\frac{1}{2}\left(logG^{11} - \hat{\beta^0_{j1}}^{\,2} G^{11}\right)\right)\right\}.$$

The full conditional distribution for $\boldsymbol{\Pi}$ can be written as

$$f\left(\boldsymbol{\Pi} | \boldsymbol{\beta}, \boldsymbol{D}, \boldsymbol{G}, \boldsymbol{R}, \boldsymbol{y}\right) \propto Dirichlet\left(n_1 + 1,\ n_2 + 1, \ldots\right),$$

where n_i is the number of loci or markers for which $\boldsymbol{\delta}_j =$ "i".

The full conditional distributions for $\boldsymbol{R}$, the covariance matrix for residuals, is an inverse Wishart distribution, $W_t^{-1}\left(\mathbf{S}_e + \boldsymbol{e}'\boldsymbol{e}, \nu_e + n\right)$, where $\boldsymbol{e}$ is the $n \times t$ matrix for residuals whose ith row is $\boldsymbol{e}_i'$. The full conditional distribution for $\boldsymbol{G}$, the covariance matrix for β_j, is an inverse Wishart distribution, $W_t^{-1}\left(\mathbf{S}_\beta + \boldsymbol{\beta}'\boldsymbol{\beta}, \nu_\beta + p\right)$, where $\boldsymbol{\beta}$ is the $p \times t$ matrix whose ith row is $\boldsymbol{\beta}_i'$.

5.3.2.3 Gibbs sampler II for multi-trait BayesCΠ

The Gibbs sampler above requires that all 2^t outcomes for $\boldsymbol{\delta}_j$ have non-zero prior probabilities, i.e. none of Π_i can be zero. If some Π_i are zero, the markov chain generated from Gibbs sampler I may not be irreducible. Further, if some particular Π_i are near zero, the chain might exhibit mixing problems. Another more general Gibbs sampler that does not require all Π_i to be non-zero and may exhibit improved mixing is proposed below.

The full conditional distributions of $\boldsymbol{\beta}_j$, $\boldsymbol{\delta}_j$, $\boldsymbol{\Pi}$, $\mathbf{G}$, $\mathbf{R}$ for Gibbs sampler II, whose derivations are in the Appendix, are given below.

Let $\boldsymbol{\theta}$ denote all other parameters except $\boldsymbol{\beta}_j$ and $\boldsymbol{\delta}_j$, then our sampling scheme can be written as

$$f\left(\boldsymbol{\beta}_j, \boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) = f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) f\left(\boldsymbol{\beta}_j \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right).$$

The full conditional distribution of $\boldsymbol{\beta}_j$ is

$$f\left(\boldsymbol{\beta}_j \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right) \propto N\left(\boldsymbol{C}_j^{-1}\boldsymbol{r}_j, \boldsymbol{C}_j^{-1}\right),$$

where $\boldsymbol{C}_j = \boldsymbol{D}_j^{'}\boldsymbol{R}^{-1}\boldsymbol{D}_j \sum_{i=1}^{n} m_{ij}^2 + \boldsymbol{G}^{-1}$ and $\boldsymbol{r}_j^{'} = \left(\sum_{i=1}^{n} \boldsymbol{w}_i^{'} m_{ij}\right) \boldsymbol{R}^{-1}\boldsymbol{D}_j$.

The marginal full conditional probability of $\boldsymbol{\delta}_j =$ "i" is

$$\begin{aligned} & f\left(\boldsymbol{\delta}_j = \text{“}i\text{”} \mid \boldsymbol{\theta}, \boldsymbol{y}\right) \\ =& \frac{f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j = \text{“}i\text{”}, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j = \text{“}i\text{”} \mid \boldsymbol{\Pi}\right)}{\sum_{\text{“}\boldsymbol{i}\in\{\text{“}\mathbf{1}\text{”},\text{“}\mathbf{2}\text{”},\ldots,\text{“}\boldsymbol{l}\text{”}\}} f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j = \text{“}i\text{”}, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j = \text{“}i\text{”} \mid \boldsymbol{\Pi}\right)}, \end{aligned}$$

where

$$f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) = \mid \boldsymbol{C}_j^{-1} \mid^{\frac{1}{2}} exp\left\{\frac{1}{2}\boldsymbol{r}_j^{'}\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right\}.$$

This Gibbs sampler can accommodate the "restricted" multi-trait BayesCΠ that was proposed by Jia et al. (Jia and Jannink, 2012), which only allows $\boldsymbol{\delta}_j$ to be a vector of all ones or a vector of all zeros.

5.3.3 Multi-trait BayesB Model

The multi-trait BayesCΠ model proposed above can be modified to accommodate the multi-trait BayesB model. Model equation (5.1) can also be used for the multi-trait BayesB method. The differences in multi-trait BayesB method is that the prior for $\boldsymbol{\beta}_j$ is a multivariate t distribution. This is equivalent to assuming $\boldsymbol{\beta}_j$ has a multivariate normal distribution with null mean and locus-specific covariance matrix $\boldsymbol{G}_j$, which is assigned an inverse Wishart prior, $W_t^{-1}\left(\mathbf{S}_\beta, \nu_\beta\right)$.

The derivations of the full conditional distributions of parameters of interest for Gibbs samplers are shown in the Appendix. In the multi-trait BayesB model, the full conditional distributions for all parameters except $\boldsymbol{G}_j$ are similar to the multi-trait BayesCΠ model. The full conditional distribution for $\boldsymbol{G}_j$, the covariance matrix for β_j, is a inverse Wishart distribution, $W_t^{-1}\left(\mathbf{S}_\beta + \boldsymbol{\beta}_j\boldsymbol{\beta}_j^{'}, \nu_\beta + 1\right)$.

5.3.4 Data analyses

5.3.4.1 Real data

Published genotypic and deregressed phenotypic data for Loblolly Pine (Pinus Taeda L.) (Resende et al., 2012; Daetwyler et al., 2013) were used to compare single-trait and multi-trait

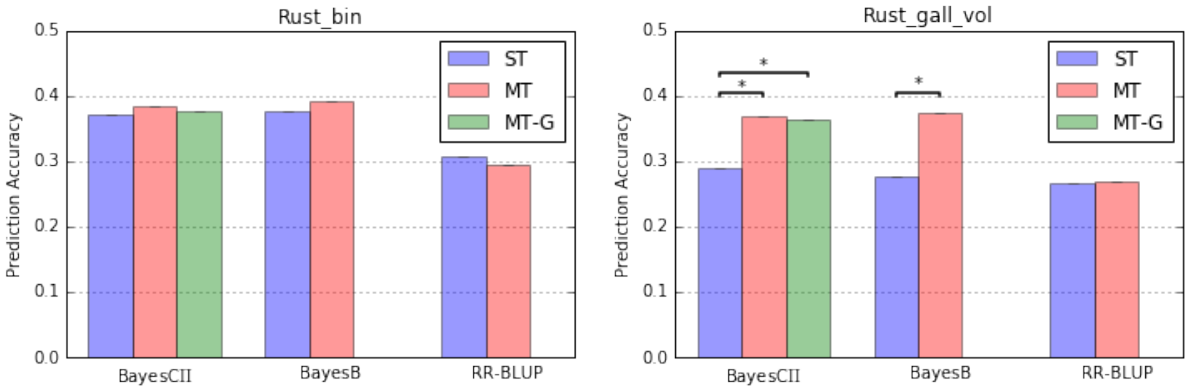

Figure 5.1 Comparison of single-trait and multi-trait methods for Rust_bin and Rust_gall_vol traits. *, indicates a statistically significant ($P < 0.01$) difference between methods.

Table 5.1 Estimation of π for alternative multi-trait BayesCΠ methods. Posterior mean of Π were given for different categories of δ. Different categories of δ are denoted as (k_1, k_2), where $k_1 = 0$ if a marker has a null effect on Rust_bin, otherwise $k_1 = 1$, and similarly for k_2 representing sampled effects for Rust_gall_vol. Combinations listed as NA do not exist in the restricted model.

	Different Categories of δ			
	$(0,0)$	$(1,1)$	$(0,1)$	$(1,0)$
MT-BayesCΠ-G	0.966	0.029	0.002	0.003
MT-BayesCΠ	0.971	0.029	NA	NA

Bayesian regression methods . Two disease traits, namely Rust_bin and Rust_gall_vol were analyzed. The reported heritability was 0.21 for Rust_bin and 0.12 for Rust_gall_vol. Loci with missing genotypes were imputed as the mean of the observed genotype covariates at that locus and loci with a missing rate >50% were excluded. After these quality control edits, 4,828 SNPs on 807 individuals with phenotypes and genotypes on both traits remained.

Prediction accuracy was calculated as the correlation between the vector of deregressed phenotypes and the vector of estimated breeding values. Cross-validation using 10 folds formed the basis for comparing these methods. Paired t tests were used for tests of significance. The general multi-trait BayesCΠ model (MT-BayesCΠ-G) were compared to a similar model where the prior for $\boldsymbol{\beta_j}$ is a multivariate normal rather than a mixture of multivariate normals (MT-

BayesC0), the more restricted multi-trait BayesCΠ proposed by Jia at al. (MT-BayesCΠ), multi-trait BayesB with known $\mathbf{\Pi}$ (MT-BayesB) and the usual single trait formulations of the mixture models (ST-BayesC0, ST-BayesCπ, ST-BayesB). The constant π used in BayesB were estimated using BayesCπ methods. Since BayesC0 is equivalent to random regression best linear unbiased prediction (RR-BLUP), ST-BayesC0 and MT-BayesC0 are denoted as ST-RR-BLUP and MT-RR-BLUP below. The prior for the residual covariance matrix $\mathbf{R}$ in all multi-trait methods was an inverse Wishart distribution, $W^{-1}\left(\begin{bmatrix} 0.003 & 0 \\ 0 & 0.003 \end{bmatrix}, 6\right)$, for which the mean of $\mathbf{R}$ is $\begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix}$. The standard deviations of diagonal elements of $\mathbf{R}$ are both 1.4×10^{-3}, and the standard deviation of off-diagonal elements of $\mathbf{R}$ are both 0. The prior for the marker effects covariance matrix $\boldsymbol{G}$ in MT-BayesCΠ and MT-BayesCΠ-G was an inverse Wishart distribution, $W^{-1}\left(\begin{bmatrix} 0.003 & 0 \\ 0 & 0.003 \end{bmatrix}, 6\right)$, for which the mean of $\boldsymbol{G}$ was $\begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix}$. The standard deviations of diagonal elements of $\mathbf{G}$ are both 1.4×10^{-3}, and the standard deviation of off-diagonal elements of $\mathbf{G}$ are both 0. The priors for the residual variance and marker effects variance in single-trait analyses were a scaled inverted chi-squared distribution with scale parameter $S^2 = 0.0005$ and degrees of freedom $\nu = 4$, for which the mean of the prior was also 0.001. Marker effect variances estimated from BayesCπ methods were used to construct the priors for marker effect variances in the BayesB methods.

5.3.4.2 Simulated data

Simulated data described below were used to investigate the value of the general multi-trait Bayesian methods under ideal conditions. The simulated genome consisted of 100 loci on each of 2 chromosomes that were in Hardy-Weinberg and linkage equilibria. All these loci were considered as QTL and used in the analyses. The QTL on the first chromosome had an effect only on trait 1 and those on the second chromosome only on trait 2.

The effects of theses QTL were simulated from a normal distribution with mean zero and

standard deviation one and then were scaled such that the genetic variance for each trait was one in a simulated population of 8,000 unrelated individuals. The phenotypes for these traits were obtained by adding independent residuals to the genetic values. Two scenarios were simulated: 1) heritabilities for both traits were 0.5; 2) heritability for trait 1 was 0.2 and for trait 2 was 0.8. The XSim package was used in the simulation (Cheng et al., 2015a).

A total of 500 individuals were used for testing, and for each training population of size N, 100 replicates of the training population were sampled from the remaining individuals. The values considered for N were 50, 100, 200, 400, 1000, 2000, 4000 or 7000. The true genetic variance and residual variance were used to compute the scale parameters for the priors of the variance components. The general multi-trait BayesCΠ model (MT-BayesCΠ-G) was compared to the more restricted multi-trait BayesCΠ (MT-BayesCΠ) using this dataset.

All analyses were performed using JWAS (Cheng et al., 2016), a publicly-available package for single-trait and multi-trait whole-genome analyses written in the freely-available Julia language.

5.4 Results

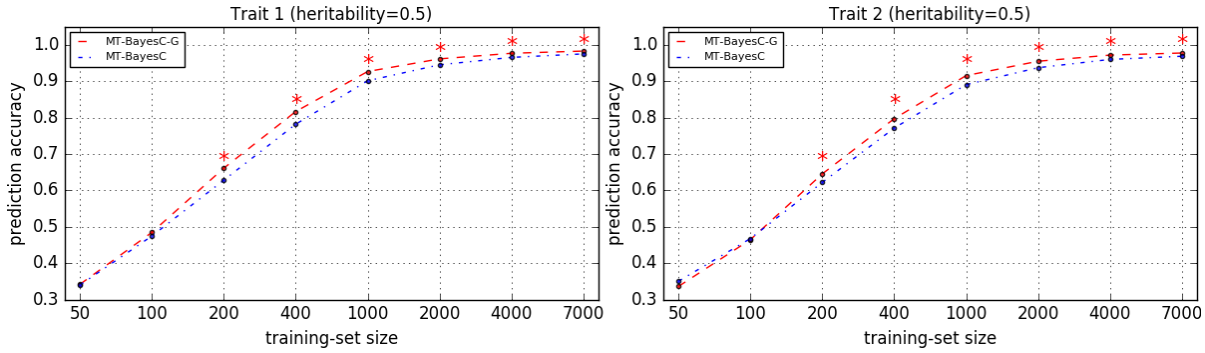

Figure 5.2 Comparison of multi-trait BayesCΠ methods under simulation scenario 1. *, indicates a statistically significant ($P < 0.01$) difference between methods.

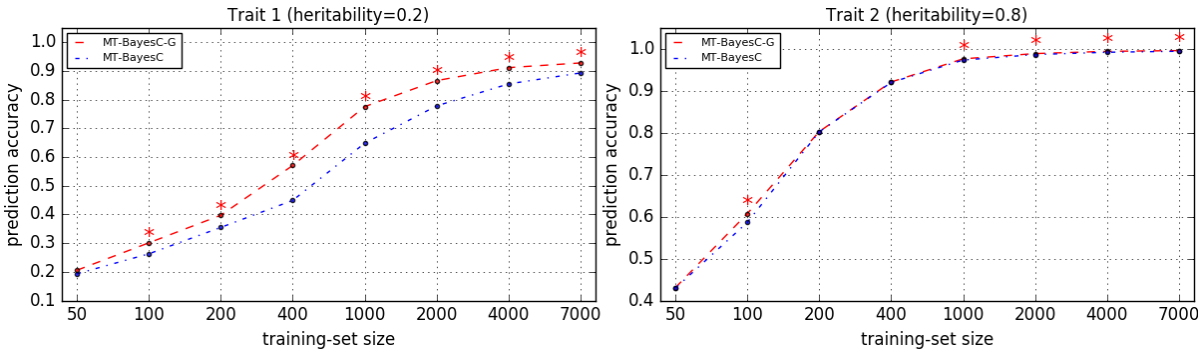

Figure 5.3 Comparison of multi-trait BayesCΠ methods under simulation scenario 2. *, indicates a statistically significant ($P < 0.01$) difference between methods.

5.4.0.1 Real data

The prediction accuracies from all methods for Rust_bin and Rust_gall_vol are in figure 5.1. The prediction accuracies from all single-trait analyses using JWAS are similar to those in (Resende et al., 2012). ST-BayesCπ showed higher prediction accuracies than ST-RR-BLUP for both traits (Resende et al., 2012). The prediction accuracies from ST-BayesB were similar to those from ST-BayesCπ, when both marker effect variances and π estimated from ST-BayesCπ were used in ST-BayesB.

The analyses of Rust_bin exhibited no significant difference between multi-trait and single-trait analyses within each method (ST-RR-BLUP versus MT-RR-BLUP; ST-BayesCπ versus MT-BayesCΠ; ST-BayesCπ versus MT-BayesCΠ-G; ST-BayesB versus MT-BayesB).

In contrast, analyses for the lower heritability Rust_gall_vol with MT-BayesCΠ-G showed significantly higher accuracies than ST-BayesCπ. MT-BayesCΠ-G and MT-BayesCΠ showed similar prediction accuracies. The posterior means of Π for both methods were shown in table 5.1. The performance of MT-BayesB were similar to MT-BayesCΠ-G, when both marker effect variances and Π estimated from MT-BayesCΠ-G were used. Similar prediction accuracies were observed in MT-RR-BLUP and ST-RR-BLUP for trait Rust_gall_vol.

5.4.0.2 Simulated data

The prediction accuracies from the general MT-BayesCΠ-G and the more restricted MT-BayesCΠ methods were compared for varying size (N) of training population. Figure 5.2 shows the prediction accuracies for the simulation scenario where heritabilities for both traits were 0.5. Figure 5.3 shows the prediction accuracies for the simulation scenario where heritabilities for trait 1 was 0.2 and for trait 2 was 0.8. For both simulation scenarios, when $N = 50$, both methods had a similar prediction accuracy. For both traits, as N increased, initially, MT-BayesCΠ-G became superior to MT-BayesCΠ, but as expected, the accuracies of these methods asymptotically converged (Karaman et al., 2016).

In most cases, the differences in accuracies for both traits were small even when they were statistically significant. However, in figure 5.3, the differences in accuracies for trait 1, for which the heritability was 0.2, were substantially large for intermediate values of N.

5.5 Discussion

5.5.1 Real data

In the single trait analyses, accuracies from ST-BayesCπ and ST-BayesB were higher than those from ST-RR-BLUP, suggesting that these two traits are influenced by a few QTL with large effects. The effect of genetic architecture on the performance of multi-trait analyses has been studied in previous simulation analyses (Jia and Jannink, 2012). Using simulated data they found that multi-trait Bayesian variable selection methods provided higher prediction accuracies than multi-trait RR-BLUP in the presence of major QTL. This observation was confirmed in our real data analyses that MT-BayesCΠ-G and MT-BayesB outperformed MT-RR-BLUP for both traits.

Significant differences between multi-trait and single-trait analyses were only observed for Rust_gall_vol within BayesCπ and BayesB methods (MT-BayesCΠ-G versus ST-BayesCπ; MT-BayesB versus ST-BayesB). MT-BayesCΠ-G and MT-BayesCΠ outperformed ST-BayesCπ for Rust_gall_vol, and the accuracy gain was 26% (from 0.287 to 0.364). The lower-heritability trait Rust_gall_vol may borrow information from the other correlated trait Rust_bin. Thus

higher prediction accuracy from MT-BayesCΠ-G were observed in trait Rust_gall_vol instead of Rust_bin. Results in (Jia and Jannink, 2012) showed no difference between MT-BayesCΠ and ST-BayesCπ because a reduced marker panel (500 markers) was used. The performance of MT-BayesB was similar to MT-BayesCΠ-G, when both marker effect variances and Π estimated from MT-BayesCΠ-G were used. Further analyses may be required to study the effects of priors on prediction accuracies in MT-BayesB.

The fact that RR-BLUP showed no improvement in multi-trait analyses suggested that benefits from MT-BayesCΠ-G may be caused by the estimation of hyper-parameter Π. In the MT-BayesCΠ-G, the mean of the posterior probability that a marker has a null effect on Rust_gall_vol was about 0.97, calculated as the summation of posterior mean of Π for categories $(0,0)$ and $(1,0)$. The posterior mean of π, the probability that a marker has a null effect, in ST-BayesCπ for Rust_gall_vol was 0.74, different from the equivalent value, 0.97, in MT-BayesCΠ-G showed above. Thus ST-BayesCπ with constant π, equal to 0.97, were performed. Prediction accuracies from ST-BayesCπ with constant $\pi = 0.97$ was 0.361, which was similar to the accuracies from MT-BayesCΠ-G. This suggests that high-heritability traits may help with variable selection in correlated low-heritability traits.

The difference between MT-BayesCΠ-G and MT-BayesCΠ is that MT-BayesCΠ assumes a locus has an effect on all traits or none of them. This assumption of genetic architecture may not always hold. MT-BayesCΠ-G and MT-BayesCΠ, however, showed similar prediction accuracies. This can be explained by the estimation of Π in MT-BayesCΠ-G and MT-BayesCΠ in table 5.1. The posterior probability means for $(0,1)$ and $(1,0)$ were almost zero in MT-BayesCΠ-G and for $(0,0)$ and $(1,1)$ are similar in MT-BayesCΠ-G and MT-BayesCΠ, suggesting that the assumption of genetic architecture for MT-BayesCΠ is valid for these two traits.

5.5.2 Simulated data

To study the advantage of the general MT-BayesCΠ-G over the more restricted MT-BayesCΠ, we simulated bivariate data where each locus had an effect on only one of the traits. In MT-BayesCΠ, if a locus has an effect on one of the traits, that locus is included in the model for all traits. So in the simulated data, MT-BayesCΠ would ideally include all loci

in the model for both traits. Thus for the trait that had heritability 0.2, the contribution of noise to the prediction from the loci on chromosome 2, which had no effect on this trait, is large relative to the signal from loci on chromosome 1. In contrast, the general variable selection in MT-BayesCΠ-G allows loci on chromosome 2, which have no effect on trait 1, to be excluded from the model for trait 1. Thus when sufficient data were available for variable selection to exclude loci on chromosome 2 for trait 1, MT-BayesCΠ-G showed a substantial advantage over MT-BayesCΠ. On the other hand, for the trait with heritability 0.8, the contribution of noise to the prediction from the loci on chromosome 1, which had no effect on this trait, is small relative to the signal from loci on chromosome 2. Thus MT-BayesCΠ-G and MT-BayesCΠ had similar accuracies. As the training population size increased, the contribution of noise to the prediction of a trait from loci, which had no effect on this trait, vanished even when the heritability was low. This was observed for both traits in both figure 5.2 and 5.3.

5.5.3 Priors

In practice, genetic variances from previous conventional analyses are usually used to construct priors for marker effect variances. For single trait analyses, under some assumptions, it can be shown that the marker effect variance σ^2_α can be obtained as

$$\sigma^2_\alpha = \frac{\sigma^2_g}{(1-\pi)\sum 2p_j(1-p_j)}, \tag{5.2}$$

where σ^2_g is the genetic variance, p_j is the allele frequency for locus j and π is the probability that a marker has a null effect (Habier et al., 2007; Gianola et al., 2009b; Fernando and Garrick, 2013b). Following a similar strategy, the marker effect covariance matrix $\mathbf{G}$ in a two-trait analysis can be obtained as

$$\mathbf{G} = \frac{1}{\sum 2p_j(1-p_j)} \begin{bmatrix} \frac{Q_{11}}{p(\boldsymbol{\delta}=(1,1))+p(\boldsymbol{\delta}=(1,0))} & \frac{Q_{12}}{p(\boldsymbol{\delta}=(1,1))} \\ \frac{Q_{21}}{p(\boldsymbol{\delta}=(1,1))} & \frac{Q_{22}}{p(\boldsymbol{\delta}=(1,1))+p(\boldsymbol{\delta}=(0,1))} \end{bmatrix}, \tag{5.3}$$

where $\mathbf{Q} = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix}$ is the genetic covariance matrix and $p(\boldsymbol{\delta}=(0,1))$, $p(\boldsymbol{\delta}=(1,0))$, $p(\boldsymbol{\delta}=(1,1))$ are the probability a marker has null effects on the first trait but not the second

trait, on the second trait but not the first trait and on no traits. Thus the probability that a marker has an effect on the first trait can be obtained as $p(\boldsymbol{\delta} = (1,1)) + p(\boldsymbol{\delta} = (1,0))$, which is the denominator of the upper left element in (5.3). This strategy relating genetic covariance matrix to marker effect covariance matrix can also be used for analyses with more than two traits. Note that positive definite matrix $\mathbf{Q}$ may result in negative definite matrix $\mathbf{G}$ using (5.3), especially when the prior for the probability a marker has null effects violates the truth. In this case, the diagonal elements of $\mathbf{G}$, which are the marker effect variances for different traits, can be obtained using (5.2), where π may be estimated from previous single-trait analyses, and the off-diagonal elements of $\mathbf{G}$ may be set to zero to guarantee positive definiteness of $\mathbf{G}$.

5.5.4 Summary and conclusions

In regard to a single trait, a locus either has an effect, or it does not. Hence, the scalar parameter π (and its complement $1 - \pi$) completely defines this circumstance. In a multi-trait setting, it is conceivable that loci that influence one trait, may or may not influence other traits. In that circumstance, a vector Π is required to define the geneitc architecture. The number of parameters that constitute the vector Π is 2^t which grows rapidly with the number of traits. In most cases, the researcher will have little or no knowledge of the likely extent of pleiotropy of loci that influence two traits, other than knowing or having an estimate of the genetic covariance. There are two simple ways to reduce this complexity in priors.

First, one can assume as did Jia *et al*. that in the context of variable selection a locus should be selected for all of the traits or selected for none of the traits, reducing the required probabilities to being analogous to the single trait π and $(1 - \pi)$. This approach has the advantage of simplicity, but the disadvantage that many effects might need to be estimated for loci that have no effect on a trait, and this may erode the accuracy of prediction. This should not be a problem for asymptotically large datasets, as in that case the fitted locus effects should converge to zero for those traits not influenced by that locus.

A second simple way to accommodate the multiple trait circumstance is to assume the 2^t parameters can be derived from t trait-specific parameters. However, when the probability that a single trait locus has an effect is small for each of two or more traits, the pair-wise probability

that a locus affects all the traits will be the product of those small probabilities, making it very difficult for loci to enter the model for all traits simultaneously.

The better way to solve this problem is to use a hyper-parameter Π that completely defines the alternative models that are required to capture all the alternative forms of genetic architecture. We have shown here how this can be done, with two alternative Gibbs sampling strategies. One involves single-site sampling for one locus and trait at a time. The other samples all the alternative combinations of effects for one locus considering all traits simultaneously. We have shown that both are practical with real data and can result in improved accuracies of prediction in certain circumstances in terms of genetic architecture and size of dataset.

Many researchers are interested in pleiotropy and would therefore want to know which loci affect which traits, from a purely biological perspective. Practitioners are often interested in "breaking" the genetic correlation, by selecting parents to give a favorable selection response in respect to multiple trait consequences. In either of these circumstances, with intermediate- rather than asymptotically-large datasets, we believe the methods described here and available in the freely-available JWAS package offer real promise.

Author's contributions

HC, KK, JZ, DG, RF contributed to the development of the statistical methods. HC wrote the program code and conducted the analyses. The manuscript was prepared by HC, RF and DG. All authors read and approved the final manuscript.

5.6 Appendix

5.6.1 Gibbs sampler algorithm for multi-trait BayesCΠ

5.6.1.1 Single-site Gibbs sampler for multi-trait BayesCΠ

The full conditional distribution of β_{j1} can be written as

$$
\begin{aligned}
& f\left(\beta_{j1} \mid \delta_{j1}, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}_{-j1}, \boldsymbol{G}, \boldsymbol{R}, \boldsymbol{y}\right) \\
& \propto f\left(\boldsymbol{y} \mid \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{D}, \boldsymbol{G}, \boldsymbol{R}\right) f\left(\beta_{j1}, \boldsymbol{\beta}_{j2} \mid \boldsymbol{G}\right) \\
& \propto exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)^{'}\boldsymbol{R}^{-1}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)\right] exp\left(-\frac{1}{2}\boldsymbol{\beta}_j^{'}\boldsymbol{G}^{-1}\boldsymbol{\beta}_j\right),
\end{aligned}
$$

where $\boldsymbol{w}_i = \boldsymbol{y}_i - \boldsymbol{\mu}_i - \sum_{j' \neq j} m_{ij'}\boldsymbol{D}_{j'}\boldsymbol{\beta}_{j'}$. Further, by dropping factors that do not involve β_{j1},

$$
\begin{aligned}
& f\left(\beta_{j1} \mid \delta_{j1}, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}_{-j1}, \boldsymbol{G}, \boldsymbol{R}, \boldsymbol{y}\right) \\
& \propto exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j^{'}\left(\boldsymbol{D}_j^{'}\boldsymbol{R}^{-1}\boldsymbol{D}_j\sum_{i=1}^{n}m_{ij}^2 + \boldsymbol{G}^{-1}\right)\boldsymbol{\beta}_j - 2\sum_{i=1}^{n}\boldsymbol{w}_i^{'}m_{ij}\boldsymbol{R}^{-1}\boldsymbol{D}_j\boldsymbol{\beta}_j\right]\right\} \\
& \propto exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j^{'}\boldsymbol{C}_j\boldsymbol{\beta}_j - 2\boldsymbol{r}_j^{'}\boldsymbol{\beta}_j\right]\right\} \\
& \propto exp\left\{-\frac{1}{2}\left[\begin{bmatrix}\beta_{j1} & \boldsymbol{\beta}_{j2}^{'}\end{bmatrix}\begin{bmatrix}C_{j,11} & \boldsymbol{C}_{j,12} \\ \boldsymbol{C}_{j,21} & \boldsymbol{C}_{j,22}\end{bmatrix}\begin{bmatrix}\beta_{j1} \\ \boldsymbol{\beta}_{j2}\end{bmatrix} - 2\begin{bmatrix}r_{j1} & \boldsymbol{r}_{j2}^{'}\end{bmatrix}\begin{bmatrix}\beta_{j1} \\ \boldsymbol{\beta}_{j2}\end{bmatrix}\right]\right\} \\
& \propto exp\left\{-\frac{1}{2}\left(C_{j,11}\beta_{j1}^2 + \left(2\boldsymbol{C}_{j,12}\boldsymbol{\beta}_{j2} - 2r_{j1}\right)\beta_{j1}\right)\right\} \\
& \propto exp\left\{-\frac{C_{j,11}}{2}\left(\beta_{j1} + \left(\boldsymbol{C}_{j,12}\boldsymbol{\beta}_{j2} - r_{j1}\right)C_{j,11}^{-1}\right)^2\right\} \\
& \propto N\left(C_{j,11}^{-1}\left(r_{j1} - \boldsymbol{C}_{j,12}\boldsymbol{\beta}_{j2}\right), C_{j,11}^{-1}\right) \\
& \propto N\left(\hat{\beta_{j1}}, C_{j,11}^{-1}\right)
\end{aligned}
$$

where $\boldsymbol{C}_j = \boldsymbol{D}_j^{'}\boldsymbol{R}^{-1}\boldsymbol{D}_j\sum_{i=1}^{n}m_{ij}^2 + \boldsymbol{G}^{-1}$ and $\boldsymbol{r}_j^{'} = \left(\sum_{i=1}^{n}\boldsymbol{w}_i^{'}m_{ij}\right)\boldsymbol{R}^{-1}\boldsymbol{D}_j$.

Note that when $\delta_{j1} = 0$,

$$\begin{aligned}
\mathbf{C}_j &= \begin{bmatrix} C^0_{j,11} & \boldsymbol{C}^0_{j,12} \\ \boldsymbol{C}^0_{j,21} & \boldsymbol{C}^0_{j,22} \end{bmatrix} \\
&= \begin{bmatrix} G^{11} & \boldsymbol{G}^{12} \\ \boldsymbol{G}^{21} & \boldsymbol{G}^{22} + \boldsymbol{D}^{'}_{j2}\boldsymbol{R}^{22}\boldsymbol{D}_{j2}\sum_{i=1}^n m_{ij}^2 \end{bmatrix} \\
\mathbf{r}^{'}_j &= \begin{bmatrix} r^0_{j1} & \boldsymbol{r}^{0\,'}_{j2} \end{bmatrix} \\
&= \begin{bmatrix} 0 & \left(\sum_{i=1}^n \boldsymbol{w}^{'}_i m_{ij}\right) \begin{bmatrix} \boldsymbol{R}^{12} \\ \boldsymbol{R}^{22} \end{bmatrix} \boldsymbol{D}_{j2} \end{bmatrix}
\end{aligned}$$

When $\delta_{j1} = 1$,

$$\begin{aligned}
\mathbf{C}_j &= \begin{bmatrix} C^1_{j,11} & \boldsymbol{C}^1_{j,12} \\ \boldsymbol{C}^1_{j,21} & \boldsymbol{C}^1_{j,22} \end{bmatrix} \\
&= \begin{bmatrix} G^{11} + R^{11}\sum_{i=1}^n m_{ij}^2 & \boldsymbol{G}^{12} + \boldsymbol{R}^{12}\boldsymbol{D}_{j2}\sum_{i=1}^n m_{ij}^2 \\ \boldsymbol{G}^{21} + \boldsymbol{D}^{'}_{j2}\boldsymbol{R}^{21}\sum_{i=1}^n m_{ij}^2 & \boldsymbol{G}^{22} + \boldsymbol{D}^{'}_{j2}\boldsymbol{R}^{22}\boldsymbol{D}_{j2}\sum_{i=1}^n m_{ij}^2 \end{bmatrix} \\
\mathbf{r}^{'}_j &= \begin{bmatrix} r^1_{j1} & \boldsymbol{r}^{1\,'}_{j2} \end{bmatrix} \\
&= \begin{bmatrix} \left(\sum_{i=1}^n \boldsymbol{w}^{'}_i m_{ij}\right) \begin{bmatrix} R^{11} \\ \boldsymbol{R}^{21} \end{bmatrix} & \left(\sum_{i=1}^n \boldsymbol{w}^{'}_i m_{ij}\right) \begin{bmatrix} \boldsymbol{R}^{12} \\ \boldsymbol{R}^{22} \end{bmatrix} \boldsymbol{D}_{j2} \end{bmatrix}
\end{aligned}$$

Thus when $\delta_{j1} = 0$, the full conditional distribution of β_{j1} is

$$f\left(\beta_{j1} \mid \delta_{j1} = 0, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}_{-j1}, \boldsymbol{G}, \boldsymbol{R}, \boldsymbol{y}\right) \propto N\left(\hat{\beta^0_{j1}}, \left(C^0_{j,11}\right)^{-1}\right) = N\left(-\left(G^{11}\right)^{-1}\mathbf{G}^{12}\boldsymbol{\beta}_{j2}, \left(G^{11}\right)^{-1}\right).$$

When $\delta_{j1} = 1$, the full conditional distribution of β_{j1} becomes

$$f\left(\beta_{j1} \mid \delta_{j1} = 1, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}_{-j1}, \boldsymbol{G}, \boldsymbol{R}, \boldsymbol{y}\right) \propto N\left(\hat{\beta^1_{j1}}, \left(C^1_{j,11}\right)^{-1}\right) = N\left(\left(C^1_{j,11}\right)^{-1}\left(r_{j1} - \mathbf{C}^1_{j,12}\boldsymbol{\beta}_{j2}\right), \left(C^1_{j,11}\right)^{-1}\right).$$

The marginal full conditional distribution of δ_{j1} can be written as

$$\begin{aligned}
f\left(\delta_{j1} = 1 \mid \boldsymbol{\theta}, \boldsymbol{y}\right) &= \frac{f\left(\delta_{j1} = 1, \boldsymbol{\theta}, \boldsymbol{y}\right)}{\sum_{\delta_{j1}\in(0,1)} f\left(\delta_{j1}, \boldsymbol{\theta}, \boldsymbol{y}\right)} \\
&= \frac{f\left(\boldsymbol{y} \mid \delta_{j1} = 1, \boldsymbol{\theta}\right) f\left(\delta_{j1} = 1, \boldsymbol{\delta}_{j2} \mid \boldsymbol{\Pi}\right)}{\sum_{\delta_{j1}\in(0,1)} f\left(\boldsymbol{y} \mid \delta_{j1}, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)}. \\
&= \left\{1 + \frac{f\left(\boldsymbol{y} \mid \delta_{j1} = 0, \boldsymbol{\theta}\right) f\left(\delta_{j1} = 0, \boldsymbol{\delta}_{j2} \mid \boldsymbol{\Pi}\right)}{f\left(\boldsymbol{y} \mid \delta_{j1} = 1, \boldsymbol{\theta}\right) f\left(\delta_{j1} = 1, \boldsymbol{\delta}_{j2} \mid \boldsymbol{\Pi}\right)}\right\}^{-1}
\end{aligned}$$

The factor $f(\boldsymbol{y} \mid \delta_{j1}, \boldsymbol{\theta})$ can be written as

$$
\begin{aligned}
f(\boldsymbol{y} \mid \delta_{j1}, \boldsymbol{\theta}) &\propto \int f\left(\boldsymbol{y} \mid \boldsymbol{\mu}, \beta_{j1}, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}, \boldsymbol{G}, \boldsymbol{R}\right) f\left(\beta_{j1}, \boldsymbol{\beta}_{j2} \mid \boldsymbol{G}\right) d\beta_{j1} \\
&\propto \int exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)' \boldsymbol{R}^{-1}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)\right] exp\left(-\frac{1}{2}\boldsymbol{\beta}_j'\boldsymbol{G}^{-1}\boldsymbol{\beta}_j\right) d\beta_{j1} \\
&\propto exp\left\{-\frac{1}{2}\left(\sum_i \boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - 2\boldsymbol{r}_{j2}'\boldsymbol{\beta}_{j2} + \boldsymbol{\beta}_{j2}'\boldsymbol{C}_{j,22}\boldsymbol{\beta}_{j2} - \left(r_{j1} - \boldsymbol{C}_{j,12}\boldsymbol{\beta}_{j2}\right)^2 C_{j,11}^{-1}\right)\right\} \\
&\times \int exp\left[-\frac{1}{2}\left(\beta_{j1} - \hat{\beta_{j1}}\right)^2 C_{j,11}\right] d\beta_{j1} \\
&\propto \left(C_{j,11}\right)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\left(\sum_i \boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - 2\boldsymbol{r}_{j2}'\boldsymbol{\beta}_{j2} + \boldsymbol{\beta}_{j2}'\boldsymbol{C}_{j,22}\boldsymbol{\beta}_{j2} - \left(r_{j1} - \boldsymbol{C}_{j,12}\boldsymbol{\beta}_{j2}\right)^2 C_{j,11}^{-1}\right)\right\} \\
&\propto \left(C_{j,11}\right)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\left(\sum_i \boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - 2\boldsymbol{r}_{j2}'\boldsymbol{\beta}_{j2} + \boldsymbol{\beta}_{j2}'\boldsymbol{C}_{j,22}\boldsymbol{\beta}_{j2} - \hat{\beta_{j1}}^2 C_{j,11}\right)\right\}.
\end{aligned}
$$

Note that $\sum_i \boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i, \boldsymbol{r}_{j2}'\boldsymbol{\beta}_{j2}, \boldsymbol{\beta_{j2}}'\boldsymbol{C}_{j,22}\boldsymbol{\beta}_{j2}$ are same when $\delta_{j1} = 0$ or 1. Thus the ratio $\frac{f(\boldsymbol{y}|\delta_{j1}=1,\boldsymbol{\theta})}{f(\boldsymbol{y}|\delta_{j1}=0,\boldsymbol{\theta})}$ becomes

$$
\begin{aligned}
H &= \left(C_{j,11}^1\right)^{-\frac{1}{2}}\left(G^{11}\right)^{\frac{1}{2}} exp\left(-\frac{1}{2}\left(\hat{\beta_{j1}^0}^2 G^{11} - \hat{\beta_{j1}^1}^2 C_{j,11}^1\right)\right) \\
&= exp\left\{-\frac{1}{2}\left(logC_{j,11}^1 - \hat{\beta_{j1}^1}^2 C_{j,11}^1\right) - \left(-\frac{1}{2}\left(logG^{11} - \hat{\beta_{j1}^0}^2 G^{11}\right)\right)\right\}
\end{aligned}
$$

Thus the conditional probability of $\delta_{j1} = 1$ is

$$
\left\{1 + \frac{f(\boldsymbol{y} \mid \delta_{j1} = 0, \boldsymbol{\theta})\, f(\delta_{j1} = 0, \boldsymbol{\delta}_{j2} \mid \boldsymbol{\Pi})}{f(\boldsymbol{y} \mid \delta_{j1} = 1, \boldsymbol{\theta})\, f(\delta_{j1} = 1, \boldsymbol{\delta}_{j2} \mid \boldsymbol{\Pi})}\right\}^{-1} = \left\{1 + \left(\frac{\boldsymbol{\Pi}_{j0}}{\boldsymbol{\Pi}_{j1}}H\right)^{-1}\right\}^{-1},
$$

where $\boldsymbol{\Pi}_{j0} = Pr(\delta_{j1} = 0, \boldsymbol{\delta}_{j2}|\boldsymbol{\Pi})$ and $\boldsymbol{\Pi}_{j1} = Pr(\delta_{j1} = 1, \boldsymbol{\delta}_{j2}|\boldsymbol{\Pi})$.

The full conditional distribution for $\boldsymbol{\Pi}$ can be written as

$$
\begin{aligned}
f(\boldsymbol{\Pi}|\boldsymbol{\beta}, \boldsymbol{D}, \boldsymbol{G}, \boldsymbol{R}, \boldsymbol{y}) &\propto f(\boldsymbol{\delta}|\boldsymbol{\Pi})\, f(\boldsymbol{\Pi}) \\
&\propto \Pi_1^{n_1}\Pi_2^{n_2}...\Pi_l^{n_l} \\
&\propto Dirichlet(n_1 + 1, n_2 + 1, \ldots),
\end{aligned}
$$

where n_i is the number of markers with $\boldsymbol{\delta}_j =$ "i".

5.6.1.2 Joint Gibbs sampler for multi-trait BayesCΠ

Let $\boldsymbol{\theta}$ denote all other parameters except $\boldsymbol{\beta}_j$ and $\boldsymbol{\delta}_j$, then our sampling scheme can be written as

$$f\left(\boldsymbol{\beta}_j, \boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) = f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) f\left(\boldsymbol{\beta}_j \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right)$$

The marginal full conditional distribution of $\boldsymbol{\delta}_j$ can be written as

$$\begin{aligned} f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) &= \frac{f\left(\boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right)}{\sum_{\boldsymbol{\delta}_j} f\left(\boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right)} \\ &= \frac{f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)}{\sum_{\boldsymbol{\delta}_j} f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)}. \end{aligned}$$

Denote $\boldsymbol{w}_i = \boldsymbol{y}_i - \boldsymbol{\mu}_i - \sum_{j' \neq j} m_{ij'} \boldsymbol{D}_{j'} \boldsymbol{\beta}_{j'}$, then

$$\begin{aligned} & f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) \\ & \propto \int f\left(\boldsymbol{y} \mid \boldsymbol{\beta}, \boldsymbol{D}, \boldsymbol{R}\right) f\left(\boldsymbol{\beta}_j \mid \boldsymbol{G}\right) d\boldsymbol{\beta}_j \\ & \propto \int exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)' \boldsymbol{R}^{-1}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)\right] exp\left(-\frac{1}{2}\boldsymbol{\beta}_j'\boldsymbol{G}^{-1}\boldsymbol{\beta}_j\right) d\boldsymbol{\beta}_j \\ & \propto \int exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j'\left(\boldsymbol{D}_j'\boldsymbol{R}^{-1}\boldsymbol{D}_j\sum_{i=1}^{n} m_{ij}^2 + \boldsymbol{G}^{-1}\right)\boldsymbol{\beta}_j - 2\sum_{i=1}^{n}\boldsymbol{w}_i' m_{ij}\boldsymbol{R}^{-1}\boldsymbol{D}_j\boldsymbol{\beta}_j + \sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i\right]\right\} d\boldsymbol{\beta}_j \\ & \propto \int exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j'\boldsymbol{C}_j\boldsymbol{\beta}_j - 2\boldsymbol{r}_j'\boldsymbol{\beta}_j + \sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i\right]\right\} d\boldsymbol{\beta}_j \\ & \propto \int exp\left\{-\frac{1}{2}\left[\left(\boldsymbol{\beta}_j' - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\right)\boldsymbol{C}_j\left(\boldsymbol{\beta}_j - \boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right) + \sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right]\right\} d\boldsymbol{\beta}_j \\ & \propto exp\left\{-\frac{1}{2}\left[\sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right]\right\} \\ & \times \mid \boldsymbol{C}_j^{-1} \mid^{\frac{1}{2}} \int \mid \boldsymbol{C}_j^{-1} \mid^{-\frac{1}{2}} exp\left[-\frac{1}{2}\left(\boldsymbol{\beta}_j' - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\right)\boldsymbol{C}_j\left(\boldsymbol{\beta}_j - \boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right)\right] d\boldsymbol{\beta}_j \\ & \propto \mid \boldsymbol{C}_j^{-1} \mid^{\frac{1}{2}} exp\left\{-\frac{1}{2}\left[\sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right]\right\}, \end{aligned}$$

where $\boldsymbol{C}_j = \boldsymbol{D}_j'\boldsymbol{R}^{-1}\boldsymbol{D}_j\sum_{i=1}^{n} m_{ij}^2 + \boldsymbol{G}^{-1}$ and $\boldsymbol{r}_j' = \left(\sum_{i=1}^{n}\boldsymbol{w}_i' m_{ij}\right)\boldsymbol{R}^{-1}\boldsymbol{D}_j$.

Note that $\sum_i \boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i$ is same for different $\boldsymbol{\delta}_j$. Thus the marginal full conditional distribution of $\boldsymbol{\delta}_j$ can be written as

$$f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) = \frac{f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)}{\sum_{\boldsymbol{\delta}_j} f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)},$$

where

$$f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) \propto \mid \boldsymbol{C}_j^{-1} \mid^{\frac{1}{2}} exp\left\{\frac{1}{2}\boldsymbol{r}_j^{'}\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right\}.$$

The full conditional distribution of $\boldsymbol{\beta}_j$ is

$$\begin{aligned}
&f\left(\boldsymbol{\beta}_j \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right) \\
&\propto exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)^{'}\boldsymbol{R}^{-1}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)\right] exp\left(-\frac{1}{2}\boldsymbol{\beta}_j^{'}\boldsymbol{G}^{-1}\boldsymbol{\beta}_j\right), \\
&\propto exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j^{'}\left(\boldsymbol{D}_j^{'}\boldsymbol{R}^{-1}\boldsymbol{D}_j\sum_{i=1}^{n}m_{ij}^2 + \boldsymbol{G}^{-1}\right)\boldsymbol{\beta}_j - 2\sum_{i=1}^{n}\boldsymbol{w}_i^{'}m_{ij}\boldsymbol{R}^{-1}\boldsymbol{D}_j\boldsymbol{\beta}_j\right]\right\} \\
&\propto exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j^{'}\boldsymbol{C}_j\boldsymbol{\beta}_j - 2\boldsymbol{r}_j^{'}\boldsymbol{\beta}_j\right]\right\} \\
&\propto exp\left\{-\frac{1}{2}\left(\boldsymbol{\beta}_j^{'} - \boldsymbol{r}_j^{'}\boldsymbol{C}_j^{-1}\right)\boldsymbol{C}_j\left(\boldsymbol{\beta}_j - \boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right)\right\} \\
&\propto N\left(\boldsymbol{C}_j^{-1}\boldsymbol{r}_j, \boldsymbol{C}_j^{-1}\right)
\end{aligned}$$

5.6.2 Gibbs sampler algorithm for multi-trait BayesB

5.6.2.1 Single-site Gibbs sampler for multi-trait BayesB

For convenience, from now on let "1" denote trait k and "2" the other traits. Thus, $\boldsymbol{\beta}_j$ can be denoted as $\begin{bmatrix}\beta_{j1}\\ \boldsymbol{\beta}_{j2}\end{bmatrix}$ and $\boldsymbol{D}_j$ can be denoted as $\begin{bmatrix}\delta_{j1} & 0\\ 0 & \boldsymbol{D}_{j2}\end{bmatrix}$. The Gibbs sampler for β_{jk} and δ_{jk} is derived as below. In our sampling scheme, β_{j1} and δ_{j1} are sampled from their joint full conditional distributions, which can be written as the product of the full conditional distribution of β_{j1} given δ_{j1} and the marginal full conditional distribution of δ_j. Let $\boldsymbol{\theta}$ denote all other parameters except δ_{j1} and β_{j1}, then our sampling scheme can be written as

$$f\left(\beta_{j1}, \delta_{j1} \mid \boldsymbol{\theta}, \boldsymbol{y}\right) = f\left(\beta_{j1} \mid \delta_{j1}, \boldsymbol{\theta}, \boldsymbol{y}\right) f\left(\delta_{j1} \mid \boldsymbol{\theta}, \boldsymbol{y}\right).$$

The full conditional distribution of β_j can be written as

$$\begin{aligned}
&f\left(\beta_{j1} \mid \delta_{j1}, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}_{-j1}, \boldsymbol{G}_j, \boldsymbol{G}_{-j}, \boldsymbol{R}, \boldsymbol{y}\right) \\
&\propto f\left(\boldsymbol{y} \mid \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{D}, \boldsymbol{G}_j, \boldsymbol{G}_{-j}, \boldsymbol{R}\right) f\left(\beta_{j1}, \boldsymbol{\beta}_{j2} \mid \boldsymbol{G}_j\right) \\
&\propto exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)^{'}\boldsymbol{R}^{-1}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)\right] exp\left(-\frac{1}{2}\boldsymbol{\beta}_j^{'}\boldsymbol{G}_j^{-1}\boldsymbol{\beta}_j\right),
\end{aligned}$$

where $\boldsymbol{w}_i = \boldsymbol{y}_i - \boldsymbol{\mu}_i - \sum_{j' \neq j} m_{ij'} \boldsymbol{D}_{j'} \boldsymbol{\beta}_{j'}$. Further, by dropping factors that do not involve β_{j1},

$$
\begin{aligned}
& f\left(\beta_{j1} \mid \delta_{j1}, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}_{-j1}, G_j, \boldsymbol{G}_{-j}, \boldsymbol{R}, \boldsymbol{y}\right) \\
& \propto exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j^{'}\left(\boldsymbol{D}_j^{'}\boldsymbol{R}^{-1}\boldsymbol{D}_j \sum_{i=1}^{n} m_{ij}^2 + \boldsymbol{G}_j^{-1}\right)\boldsymbol{\beta}_j - 2\sum_{i=1}^{n} \boldsymbol{w}_i^{'} m_{ij} \boldsymbol{R}^{-1}\boldsymbol{D}_j\boldsymbol{\beta}_j\right]\right\} \\
& \propto exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j^{'}\boldsymbol{C}_j\boldsymbol{\beta}_j - 2\boldsymbol{r}_j^{'}\boldsymbol{\beta}_j\right]\right\} \\
& \propto exp\left\{-\frac{1}{2}\left[\begin{bmatrix}\beta_{j1} & \boldsymbol{\beta}_{j2}^{'}\end{bmatrix}\begin{bmatrix}C_{j,11} & \boldsymbol{C}_{j,12} \\ \boldsymbol{C}_{j,21} & \boldsymbol{C}_{j,22}\end{bmatrix}\begin{bmatrix}\beta_{j1} \\ \boldsymbol{\beta}_{j2}\end{bmatrix} - 2\begin{bmatrix}r_{j1} & \boldsymbol{r}_{j2}^{'}\end{bmatrix}\begin{bmatrix}\beta_{j1} \\ \boldsymbol{\beta}_{j2}\end{bmatrix}\right]\right\} \\
& \propto exp\left\{-\frac{1}{2}\left(C_{j,11}\beta_{j1}^2 + \left(2\boldsymbol{C}_{j,12}\boldsymbol{\beta}_{j2} - 2r_{j1}\right)\beta_{j1}\right)\right\} \\
& \propto exp\left\{-\frac{C_{j,11}}{2}\left(\beta_{j1} + \left(\boldsymbol{C}_{j,12}\boldsymbol{\beta}_{j2} - r_{j1}\right)C_{j,11}^{-1}\right)^2\right\} \\
& \propto N\left(C_{j,11}^{-1}\left(r_{j1} - \boldsymbol{C}_{j,12}\boldsymbol{\beta}_{j2}\right), C_{j,11}^{-1}\right) \\
& \propto N\left(\hat{\beta_{j1}}, C_{j,11}^{-1}\right)
\end{aligned}
$$

where $\boldsymbol{C}_j = \boldsymbol{D}_j^{'}\boldsymbol{R}^{-1}\boldsymbol{D}_j \sum_{i=1}^{n} m_{ij}^2 + \boldsymbol{G}_j^{-1}$ and $\boldsymbol{r}_j^{'} = \left(\sum_{i=1}^{n} \boldsymbol{w}_i^{'} m_{ij}\right)\boldsymbol{R}^{-1}\boldsymbol{D}_j$.

Note that when $\delta_{j1} = 0$,

$$
\begin{aligned}
\mathbf{C}_j &= \begin{bmatrix} G_j^{11} & \boldsymbol{G}_j^{12} \\ \boldsymbol{G}_j^{21} & \boldsymbol{G}_j^{22} + \boldsymbol{D}_{j2}^{'}\boldsymbol{R}^{22}\boldsymbol{D}_{j2}\sum_{i=1}^{n} m_{ij}^2 \end{bmatrix} \\
\mathbf{r}_j^{'} &= \begin{bmatrix} 0 & \left(\sum_{i=1}^{n} \boldsymbol{w}_i^{'} m_{ij}\right)\begin{bmatrix}\boldsymbol{R}^{12} \\ \boldsymbol{R}^{22}\end{bmatrix}\boldsymbol{D}_{j2} \end{bmatrix}
\end{aligned}
$$

When $\delta_{j1} = 1$,

$$
\begin{aligned}
\mathbf{C}_j &= \begin{bmatrix} C_{j,11}^1 & \boldsymbol{C}_{j,12}^1 \\ \boldsymbol{C}_{j,21}^1 & \boldsymbol{C}_{j,22}^1 \end{bmatrix} \\
&= \begin{bmatrix} G_j^{11} + R^{11}\sum_{i=1}^{n} m_{ij}^2 & \boldsymbol{G}_j^{12} + \boldsymbol{R}^{12}\boldsymbol{D}_{j2}\sum_{i=1}^{n} m_{ij}^2 \\ \boldsymbol{G}_j^{21} + \boldsymbol{D}_{j2}^{'}\boldsymbol{R}^{21}\sum_{i=1}^{n} m_{ij}^2 & \boldsymbol{G}_j^{22} + \boldsymbol{D}_{j2}^{'}\boldsymbol{R}^{22}\boldsymbol{D}_{j2}\sum_{i=1}^{n} m_{ij}^2 \end{bmatrix} \\
\mathbf{r}_j^{'} &= \begin{bmatrix} r_{j1}^1 & \boldsymbol{r}_{j2}^{1'} \end{bmatrix} \\
&= \begin{bmatrix} \left(\sum_{i=1}^{n} \boldsymbol{w}_i^{'} m_{ij}\right)\begin{bmatrix}R^{11} \\ \boldsymbol{R}^{21}\end{bmatrix} & \left(\sum_{i=1}^{n} \boldsymbol{w}_i^{'} m_{ij}\right)\begin{bmatrix}\boldsymbol{R}^{12} \\ \boldsymbol{R}^{22}\end{bmatrix}\boldsymbol{D}_{j2} \end{bmatrix}
\end{aligned}
$$

Thus when $\delta_{j1} = 0$, the full conditional distribution of β_{j1} is

$$f\left(\beta_{j1} \mid \delta_{j1} = 0, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}_{-j1}, G_j, \boldsymbol{G}_{-j}, \boldsymbol{R}, \boldsymbol{y}\right) \propto N\left(-\left(G_j^{11}\right)^{-1} \mathbf{G}_j^{12} \boldsymbol{\beta}_{j2}, \left(G_j^{11}\right)^{-1}\right).$$

When $\delta_{j1} = 1$, the full conditional distribution of β_{j1} becomes

$$f\left(\beta_{j1} \mid \delta_{j1} = 1, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}_{-j1}, G_j, \boldsymbol{G}_{-j}, \boldsymbol{R}, \boldsymbol{y}\right) \propto N\left(C_{j,11}^{1-1}\left(r_{j1} - \mathbf{C}_{j,12}^{1} \boldsymbol{\beta}_{j2}\right), \mathrm{C}_{j,11}^{1-1}\right).$$

The marginal full conditional distribution of δ_{j1} can be written as

$$\begin{aligned} f\left(\delta_{j1} = 1 \mid \boldsymbol{\theta}, \boldsymbol{y}\right) &= \frac{f\left(\delta_{j1}, \boldsymbol{\theta}, \boldsymbol{y}\right)}{\sum_{\delta_{j1} \in (0,1)} f\left(\delta_{j1}, \boldsymbol{\theta}, \boldsymbol{y}\right)} \\ &= \frac{f\left(\boldsymbol{y} \mid \delta_{j1} = 1, \boldsymbol{\theta}\right) f\left(\delta_{j1} = 1, \boldsymbol{\delta}_{j2} \mid \boldsymbol{\Pi}\right)}{\sum_{\delta_{j1} \in (0,1)} f\left(\boldsymbol{y} \mid \delta_{j1}, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)}. \\ &= \left\{1 + \frac{f\left(\boldsymbol{y} \mid \delta_{j1} = 0, \boldsymbol{\theta}\right) f\left(\delta_{j1} = 0, \boldsymbol{\delta}_{j2} \mid \boldsymbol{\Pi}\right)}{f\left(\boldsymbol{y} \mid \delta_{j1} = 1, \boldsymbol{\theta}\right) f\left(\delta_{j1} = 1, \boldsymbol{\delta}_{j2} \mid \boldsymbol{\Pi}\right)}\right\}^{-1} \end{aligned}$$

The factor $f\left(\boldsymbol{y} \mid \delta_{j1}, \boldsymbol{\theta}\right)$ can be written as

$$\begin{aligned} f\left(\boldsymbol{y} \mid \delta_{j1}, \boldsymbol{\theta}\right) &\propto \int f\left(\boldsymbol{y} \mid \boldsymbol{\mu}, \beta_{j1}, \boldsymbol{\beta}_{-j1}, \boldsymbol{D}, \boldsymbol{G}, \boldsymbol{R}\right) f\left(\beta_{j1}, \boldsymbol{\beta}_{j2} \mid \boldsymbol{G}_j\right) d\beta_{j1} \\ &\propto \int exp\left[-\frac{1}{2} \sum_{i=1}^{n} \left(\boldsymbol{w}_i - m_{ij} \boldsymbol{D}_j \boldsymbol{\beta}_j\right)' R^{-1} \left(\boldsymbol{w}_i - m_{ij} \boldsymbol{D}_j \boldsymbol{\beta}_j\right)\right] exp\left(-\frac{1}{2} \boldsymbol{\beta}_j' \boldsymbol{G}_j^{-1} \boldsymbol{\beta}_j\right) d\beta_{j1} \\ &\propto exp\left\{-\frac{1}{2}\left(\sum_i \boldsymbol{w}_i' R^{-1} \boldsymbol{w}_i - 2\boldsymbol{r}_{j2}' \boldsymbol{\beta}_{j2} + \boldsymbol{\beta}_{j2}' \boldsymbol{C}_{j,22} \boldsymbol{\beta}_{j2} - \left(r_{j1} - \boldsymbol{C}_{j,12} \boldsymbol{\beta}_{j2}\right)^2 C_{j,11}^{-1}\right)\right\} \\ &\times \int exp\left[-\frac{1}{2}\left(\beta_{j1} - \hat{\beta_{j1}}\right)^2 C_{j,11}\right] d\beta_{j1} \\ &\propto \left(C_{j,11}\right)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\left(\sum_i \boldsymbol{w}_i' R^{-1} \boldsymbol{w}_i - 2\boldsymbol{r}_{j2}' \boldsymbol{\beta}_{j2} + \boldsymbol{\beta}_{j2}' \boldsymbol{C}_{j,22} \boldsymbol{\beta}_{j2} - \left(r_{j1} - \boldsymbol{C}_{j,12} \boldsymbol{\beta}_{j2}\right)^2 C_{j,11}^{-1}\right)\right\} \\ &\propto \left(C_{j,11}\right)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\left(\sum_i \boldsymbol{w}_i' R^{-1} \boldsymbol{w}_i - 2\boldsymbol{r}_{j2}' \boldsymbol{\beta}_{j2} + \boldsymbol{\beta}_{j2}' \boldsymbol{C}_{j,22} \boldsymbol{\beta}_{j2} - \hat{\beta_{j1}}^2 C_{j,11}\right)\right\}. \end{aligned}$$

Note that $\sum_i \boldsymbol{w}_i' R^{-1} \boldsymbol{w}_i, \boldsymbol{r}_{j2}' \boldsymbol{\beta}_{j2}, \boldsymbol{\beta_{j2}}' \boldsymbol{C}_{j,22} \boldsymbol{\beta}_{j2}$ are same when $\delta_{j1} = 0$ or 1. Thus the ratio $\frac{f(\boldsymbol{y}|\delta_{j1}=1,\boldsymbol{\theta})}{f(\boldsymbol{y}|\delta_{j1}=0,\boldsymbol{\theta})}$ becomes

$$\begin{aligned} H &= \left(C_{j,11}^{1}\right)^{-\frac{1}{2}} \left(G_j^{11}\right)^{\frac{1}{2}} exp\left(-\frac{1}{2}\left(\hat{\beta_{j1}^{0}}^2 G_j^{11} - \hat{\beta_{j1}^{1}}^2 C_{j,11}^{1}\right)\right) \\ &= exp\left\{-\frac{1}{2}\left(logC_{j,11}^{1} - \hat{\beta_{j1}^{1}}^2 C_{j,11}^{1}\right) - \left(-\frac{1}{2}\left(logG_j^{11} - \hat{\beta_{j1}^{0}}^2 G_j^{11}\right)\right)\right\} \end{aligned}$$

Thus the conditional probability of $\delta_{j1} = 1$ is

$$\left\{1 + \frac{f\left(\boldsymbol{y} \mid \delta_{j1} = 0, \boldsymbol{\theta}\right) f\left(\delta_{j1} = 0, \boldsymbol{\delta}_{j2} \mid \Pi_1, \Pi_{2\ldots}\right)}{f\left(\boldsymbol{y} \mid \delta_{j1} = 1, \boldsymbol{\theta}\right) f\left(\delta_{j1} = 1, \boldsymbol{\delta}_{j2} \mid \Pi_1, \Pi_{2\ldots}\right)}\right\}^{-1} = \left\{1 + \left(\frac{\boldsymbol{\Pi}_{j0}}{\boldsymbol{\Pi}_{j1}} H\right)^{-1}\right\}^{-1},$$

where $\boldsymbol{\Pi}_{j0} = Pr\left(\delta_{j1} = 0, \boldsymbol{\delta}_{j2} | \boldsymbol{\Pi}\right)$ and $\boldsymbol{\Pi}_{j1} = Pr\left(\delta_{j1} = 1, \boldsymbol{\delta}_{j2} | \boldsymbol{\Pi}\right)$.

5.6.2.2 Joint Gibbs sampler for multi-trait BayesB

Let $\boldsymbol{\theta}$ denote all other parameters except $\boldsymbol{\beta}_j$ and $\boldsymbol{\delta}_j$, then our sampling scheme can be written as

$$f\left(\boldsymbol{\beta}_j, \boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) = f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) f\left(\boldsymbol{\beta}_j \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right)$$

The marginal full conditional distribution of $\boldsymbol{\delta}_j$ can be written as

$$\begin{aligned} f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) &= \frac{f\left(\boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right)}{\sum_{\boldsymbol{\delta}_j} f\left(\boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right)} \\ &= \frac{f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)}{\sum_{\boldsymbol{\delta}_j} f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)}. \end{aligned}$$

Denote $\boldsymbol{w}_i = \boldsymbol{y}_i - \boldsymbol{\mu}_i - \sum_{j' \neq j} m_{ij'} \boldsymbol{D}_{j'} \boldsymbol{\beta}_{j'}$, then

$$\begin{aligned} & f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) \\ & \propto \int f\left(\boldsymbol{y} \mid \boldsymbol{\beta}, \boldsymbol{D}, \boldsymbol{R}\right) f\left(\boldsymbol{\beta}_j \mid \boldsymbol{G}_j\right) d\boldsymbol{\beta}_j \\ & \propto \int exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)' \boldsymbol{R}^{-1}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)\right] exp\left(-\frac{1}{2}\boldsymbol{\beta}_j'\boldsymbol{G}_j^{-1}\boldsymbol{\beta}_j\right) d\boldsymbol{\beta}_j \\ & \propto \int exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j'\left(\boldsymbol{D}_j'\boldsymbol{R}^{-1}\boldsymbol{D}_j\sum_{i=1}^{n} m_{ij}^2 + \boldsymbol{G}_j^{-1}\right)\boldsymbol{\beta}_j - 2\sum_{i=1}^{n}\boldsymbol{w}_i' m_{ij}\boldsymbol{R}^{-1}\boldsymbol{D}_j\boldsymbol{\beta}_j + \sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i\right]\right\} d\boldsymbol{\beta}_j \\ & \propto \int exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j'\boldsymbol{C}_j\boldsymbol{\beta}_j - 2\boldsymbol{r}_j'\boldsymbol{\beta}_j + \sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i\right]\right\} d\boldsymbol{\beta}_j \\ & \propto \int exp\left\{-\frac{1}{2}\left[\left(\boldsymbol{\beta}_j' - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\right)\boldsymbol{C}_j\left(\boldsymbol{\beta}_j - \boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right) + \sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right]\right\} d\boldsymbol{\beta}_j \\ & \propto exp\left\{-\frac{1}{2}\left[\sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right]\right\} \\ & \times \mid \boldsymbol{C}_j^{-1} \mid^{\frac{1}{2}} \int \mid \boldsymbol{C}_j^{-1} \mid^{-\frac{1}{2}} exp\left[-\frac{1}{2}\left(\boldsymbol{\beta}_j' - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\right)\boldsymbol{C}_j\left(\boldsymbol{\beta}_j - \boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right)\right] d\boldsymbol{\beta}_j \\ & \propto \mid \boldsymbol{C}_j^{-1} \mid^{\frac{1}{2}} exp\left\{-\frac{1}{2}\left[\sum_{i=1}^{n}\boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i - \boldsymbol{r}_j'\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right]\right\}, \end{aligned}$$

where $\boldsymbol{C}_j = \boldsymbol{D}_j'\boldsymbol{R}^{-1}\boldsymbol{D}_j\sum_{i=1}^{n} m_{ij}^2 + \boldsymbol{G}_j^{-1}$ and $\boldsymbol{r}_j' = \left(\sum_{i=1}^{n} w_i' m_{ij}\right)\boldsymbol{R}^{-1}\boldsymbol{D}_j$.

Note that $\sum_i \boldsymbol{w}_i'\boldsymbol{R}^{-1}\boldsymbol{w}_i$ is same for different $\boldsymbol{\delta}_j$. Thus the marginal full conditional distribution of $\boldsymbol{\delta}_j$ can be written as

$$f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\theta}, \boldsymbol{y}\right) = \frac{f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)}{\sum_{\boldsymbol{\delta}_j} f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) f\left(\boldsymbol{\delta}_j \mid \boldsymbol{\Pi}\right)},$$

where

$$f\left(\boldsymbol{y} \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}\right) \propto \mid \boldsymbol{C}_j^{-1} \mid^{\frac{1}{2}} exp\left\{\frac{1}{2}\boldsymbol{r}_j^{'}\boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right\}.$$

The full conditional distribution of $\boldsymbol{\beta}_j$ is

$$\begin{aligned}
f\left(\boldsymbol{\beta}_j \mid \boldsymbol{\delta}_j, \boldsymbol{\theta}, \boldsymbol{y}\right) &\propto exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)^{'}\boldsymbol{R}^{-1}\left(\boldsymbol{w}_i - m_{ij}\boldsymbol{D}_j\boldsymbol{\beta}_j\right)\right] exp\left(-\frac{1}{2}\boldsymbol{\beta}_j^{'}\boldsymbol{G}_j^{-1}\boldsymbol{\beta}_j\right), \\
&\propto exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j^{'}\left(\boldsymbol{D}_j^{'}\boldsymbol{R}^{-1}\boldsymbol{D}_j\sum_{i=1}^{n}m_{ij}^2 + \boldsymbol{G}_j^{-1}\right)\boldsymbol{\beta}_j - 2\sum_{i=1}^{n}\boldsymbol{w}_i^{'}m_{ij}\boldsymbol{R}^{-1}\boldsymbol{D}_j\boldsymbol{\beta}_j\right]\right\} \\
&\propto exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}_j^{'}\boldsymbol{C}_j\boldsymbol{\beta}_j - 2\boldsymbol{r}_j^{'}\boldsymbol{\beta}_j\right]\right\} \\
&\propto exp\left\{-\frac{1}{2}\left(\boldsymbol{\beta}_j^{'} - \boldsymbol{r}_j^{'}\boldsymbol{C}_j^{-1}\right)\boldsymbol{C}_j\left(\boldsymbol{\beta}_j - \boldsymbol{C}_j^{-1}\boldsymbol{r}_j\right)\right\} \\
&\propto N\left(\boldsymbol{C}_j^{-1}\boldsymbol{r}_j, \boldsymbol{C}_j^{-1}\right)
\end{aligned}$$

CHAPTER 6. COMPARISON OF ALTERNATIVE APPROACHES TO SINGLE-TRAIT GENOMIC PREDICTION USING GENOTYPED AND NON-GENOTYPED HANWOO BEEF CATTLE

Joonho Lee, Hao Cheng[1], Dorian Garrick, Bruce Golden, Jack Dekkers, Kyungdo Park,Deukhwan Lee and Rohan Fernando

A paper published in Genetics Selection Evolution

6.1 Abstract

Genomic predictions from BayesA and BayesB use training data that include animals with both phenotypes and genotypes. Single-step methodologies allow additional information from non-genotyped relatives to be included in the analysis. The single-step genomic best linear unbiased prediction (SSGBLUP) method uses a relationship matrix computed from marker and pedigree information, in which missing genotypes are imputed implicitly. Single-step Bayesian regression (SSBR) extends SSGBLUP to BayesB-like models using explicitly imputed genotypes for non-genotyped individuals.

Carcass records included 988 genotyped Hanwoo steers with 35,882 SNPs and 1438 non-genotyped steers that were measured for back-fat thickness (BFT), carcass weight (CWT), eye-muscle area (EMA), and marbling score (MAR). Single-trait pedigree-based BLUP, Bayesian methods using only genotyped individuals, SSGBLUP and SSBR methods were compared using cross-validation.

Methods using genomic information always outperformed pedigree-based BLUP when the same phenotypic data were modeled from either genotyped individuals only or both genotyped

[1] Joonho Lee and Hao Cheng contributed equally to this work

and non-genotyped individuals. For BFT and MAR, accuracies were higher with single-step methods than with BayesB, BayesC and BayesCπ. Gains in accuracy with the single-step methods ranged from +0.06 to +0.09 for BFT and from +0.05 to +0.07 for MAR. For CWT, SSBR always outperformed the corresponding Bayesian methods that used only genotyped individuals. However, although SSGBLUP incorporated information from non-genotyped individuals, prediction accuracies were lower with SSGBLUP than with BayesC (π= 0.9999) and BayesB (π= 0.98) for CWT because, for this particular trait, there was a benefit from the mixture priors of the effects of the single nucleotide polymorphisms.

Single-step methods are the preferred approaches for prediction combining genotyped and non-genotyped animals. Alternative priors allow SSBR to outperform SSGBLUP in some cases.

6.2 Introduction

Since breeding technologies using genome-wide single nucleotide polymorphism (SNP) panels became available, genomic selection was rapidly adopted for improvement of livestock and has replaced the traditionally used pedigree-based best linear unbiased prediction (PBLUP). The BayesA and BayesB hierarchical Bayesian models with locus-specific variances were proposed by Meuwissen et al. (Meuwissen et al., 2001a). BayesB can accommodate mixture models in which SNPs have zero effects with probability π (Garrick et al., 2014; Cheng et al., 2015b). When $\pi = 0$, BayesB is known as BayesA. BayesC is another widely-used Bayesian mixture model, in which a common variance is used for all SNPs instead of locus-specific variances (Kizilkaya et al., 2010), and a modification of that method known as BayesCπ treats π as an unknown parameter with a uniform prior distribution (Habier et al., 2011b).

In general, the number of individuals with genomic information is a small subset of the individuals represented in the population with pedigree and phenotypic information. "Single-step" methodologies were developed to take advantage of all pedigree, phenotypic and genomic information simultaneously (Legarra et al., 2009; Fernando et al., 2014). The single-step genomic BLUP (SSGBLUP) method uses a relationship matrix that is computed from marker and pedigree information. SSGBLUP was shown to yield a similar or higher accuracy compared to methods using only genotyped individuals (Misztal et al., 2013; Lourenco et al., 2014, 2015)

Fernando et al. (Fernando et al., 2014) proposed a class of single-step Bayesian regression methods (SSBR) to extend SSGBLUP to incorporate BayesB-like models for SNP effects (SSBR-B). Similar extensions of SSGBLUP with BayesC-like models result in SSBR-C and SSBR-Cπ. SSBR methods may promise higher prediction accuracies and provide computational benefits when many animals are genotyped. In SSGBLUP, the distribution of marker effects conditional on the variance of marker effects is assumed univariate normal, whereas in SSBR, the prior for marker effects can follow a t-distribution, a double exponential distribution or mixture distributions, which may be advantageous in some situations.

In this paper, prediction accuracies from PBLUP, BayesB, BayesC, BayesCπ, SSGBLUP and SSBR-B, SSBR-C, SSBR-Cπ were compared in terms of cross-validation accuracies.

6.3 Materials and Methods

6.3.1 Data

Young Hanwoo bulls are routinely progeny-tested in batches at the Hanwoo Improvement Center (Seo-San, Chungnam, South Korea). DNA samples were collected from steers that included the progeny-tested offspring from the 46th to 51st selection batches. SNP genotypes were determined using Illumina Bovine SNP50 v1 (50k) or Bovine HD (778k) beadchips (Illumina, CA).

Carcass records were recorded at harvest at about 24 months of age. The carcass traits used in the analyses were back-fat thickness (BFT), carcass weight (CWT), eye-muscle area (EMA), and marbling score (MAR). Park et al. (Park et al., 2013) reported heritabilities of 0.50, 0.30, 0.42 and 0.63 for BFT, CWT, EMA and MAR, respectively. Approval from the ethics committee was not required for these data since they were obtained from an existing industry database.

Of the 44k SNPs that are included on both the 50k and 778k beadchips, only autosomal SNPs with known map location were used. For quality control, SNPs that departed from the Hardy-Weinberg equilibrium ($p < 10^{-6}$) based on a Chi-square test, or had a minor allele frequency (MAF) lower than 0.01, or a missing rate higher than 0.1 were excluded from further

analysis. For the genotyped animals, SNPs with missing genotypes were imputed using Beagle 3.3 (Browning and Browning, 2007). After these quality controls, 35,882 SNPs remained for analyses.

The numerator relationship matrix (NRM) based on pedigree information and the genomic relationship matrix (GRM) based on SNP genotypes were compared. Nineteen individuals, which showed unreasonable deviations between the NRM and GRM coefficients that were probably due to errors in the DNA sampling, were eliminated. Among these 19 individuals, five appeared to have been genotyped twice with different ID since their GRM relationship coefficients were near 1.0 while their NRM relationship coefficients were close to 0. For the other 14 individuals, either the GRM relationship coefficients were near 0 while those of the NRM were near 0.25 as would be the case for mistakenly recorded half-sib individuals, or the GRM relationship coefficients were near 0.25 while those of the NRM were near 0 as would be the case for half-sibs mistakenly recorded as unrelated. After elimination of these suspect individuals, the correlation coefficient between NRM and GRM increased from 0.856 to 0.866. Finally, 988 genotyped individuals remained for genomic prediction with a mean MAF of 0.243 and mean observed heterozygosity of 0.326.

Additional carcass records for 1438 non-genotyped progeny-tested steers were collected from the 39th to the 51st selection batches for the single-step and PBLUP analyses. Ancestors of the 2426 individuals with carcass records contributed to an 11-generation pedigree file that included 9637 animals.

Genotyped individuals were assigned to five mutually exclusive groups for cross-validation. K-means clustering based on pedigree relationship coefficients was used to minimize the relatedness between training and validation sets (Saatchi et al., 2011). The five groups included 172, 280, 199, 139 and 198 individuals, respectively. Each group was used as the validation set while the remaining genotyped individuals were included in the training set. In SSGBLUP, SSBR and PBLUP with phenotypes on all animals, non-genotyped individuals were included in the training set. Phenotypes were pre-adjusted for contemporary group and age effects using multiple-trait PBLUP because animals from some progeny-test batches were assigned to different groups and because some analyses included additional non-genotyped animals from

the same batches as genotyped animals.

6.3.2 Single-trait statistical models

6.3.2.1 Pedigree-based BLUP

In these analyses, the adjusted phenotypes were modeled as:

$\mathbf{y} = \mathbf{1}\mu + \mathbf{Z}\ \mathbf{u} + \mathbf{e}$,

where $\mathbf{y}$ is a vector of adjusted phenotypic records from n_y animals, $\mathbf{1}$ is a vector of 1s, μ is the overall mean, Z is the design matrix allocating records to breeding values, $\mathbf{u}$ is the vector of breeding values, $\mathbf{e}$ is a random vector of residuals. It was assumed that u $\sim N(\mathbf{0},\ \mathbf{A}\sigma_g^2)$, where $\mathbf{A}$ is the numerator relationship matrix and σ_g^2 is the additive genetic variance. Residuals were assumed to be independently and identically distributed (iid) with null means and variance σ_e^2. Pedigree-based BLUP with phenotypes either on all animals or only on genotyped animals were referred to as PBLUP (n_y= 2426 minus validation animals) and PBLUP-G (n_y= 988 minus validation animals), respectively. Adjusted phenotypes were used to account for fixed effects in the validation set.

6.3.2.2 Bayesian methods using only genotyped animals

In these analyses, the adjusted phenotypes were modeled as:

$\mathbf{y} = \mathbf{1}\mu + \mathbf{M}_g\alpha + \mathbf{e}$,

where $\mathbf{y}$, $\mathbf{1}$ and $\mathbf{e}$ are $n_y\ \times\ 1$ vectors for $n_y = 988$ minus genotyped validation animals, μ is as defined before, $\mathbf{M}_g$ is the $n_y\ \times\ p$ matrix of SNP covariates at p loci, and α is a $p\ \times\ 1$ random vector of allele substitution effects. A flat prior was used for μ. The prior for $\mathbf{e}$ was $\mathbf{e}|\sigma_e^2 \sim\ N(0, \mathbf{I}\sigma_e^2)$ with $(\sigma_e^2|\nu_e, S_e^2) \sim \nu_e S_e^2 \chi^2_{\nu_e}$. Priors for SNP effects were a mixture of a point mass at zero and a t-distribution in BayesB or a mixture of a point mass at zero and a normal distribution conditional on a common variance of SNP effects in BayesC and BayesCπ methods (Garrick et al., 2014). These methods were referred to as BayesB, BayesC or BayesCπ, and ignored adjusted phenotypes on non-genotyped animals, as for PBLUP-G.

6.3.2.3 Single-step GBLUP

In the single-step GBLUP analyses, the adjusted phenotypes were modeled as:

$\mathbf{y} = \mathbf{1}\mu + \mathrm{Zu} + \mathbf{e}$,

where $\mathbf{y}$ is the vector of adjusted phenotypes as before except that it includes both genotyped and non-genotyped individuals i.e. $n_y = 2426$ minus validation animals, μ and $\mathbf{e}$ are as defined before, with residuals that are independently and identically distributed (iid) with null means and variance σ_e^2, $\mathbf{Z}$ is the design matrix allocating records to breeding values, $\mathbf{u}$ is the vector of breeding values for both genotyped and non-genotyped individuals but now $\mathbf{u} \sim N(\mathbf{0},\ \mathbf{H}\sigma_g^2)$, where:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A}_{\mathrm{ng}}\mathbf{A}_{\mathrm{gg}}^{-1}\mathbf{G}\mathbf{A}_{\mathrm{gg}}^{-1}\mathbf{A}_{\mathrm{gn}} + (\ \mathbf{A}_{\mathrm{nn}} - \mathbf{A}_{\mathrm{ng}}\mathbf{A}_{\mathrm{gg}}^{-1}\mathbf{A}_{\mathrm{gn}}\ ) & \mathbf{A}_{\mathrm{ng}}\mathbf{A}_{\mathrm{gg}}^{-1}\mathbf{G} \\ \mathbf{G}\mathbf{A}_{\mathrm{gg}}^{-1}\mathbf{A}_{\mathrm{gn}} & \mathbf{G} \end{bmatrix},$$

and $\mathbf{A}_{\mathrm{gg}}$ is the 988 order partition of the numerator relationship matrix $\mathbf{A}$ that corresponds to genotyped animals, $\mathbf{A}_{\mathrm{nn}}$ is the 8749 order partition of $\mathbf{A}$ that corresponds to non-genotyped animals, $\mathbf{A}_{\mathrm{ng}}$ or $\mathbf{A}_{\mathrm{gn}}$ are partitions of $\mathbf{A}$ of order 9637 corresponding to relationships between non-genotyped and genotyped animals or vice versa, and $\mathbf{G}$ is a GRM of order 988. We applied three methods to construct the GRM. The standard $\mathbf{G}$ was constructed as $\mathbf{G} = \frac{\mathbf{T}\mathbf{T}'}{\sum 2q_i(1-q_i)}$ (SSGBLUP-I) with $\mathbf{T}$ being the centered matrix of SNP covariates ($\mathbf{T} = \mathbf{M}_g - \frac{1}{n}\mathbf{1}\mathbf{1}'\mathbf{M}_g$), q_i representing the allele frequency of the ith SNP. This is the same $\mathbf{G}$ as previously used to compare relationship coefficients between NRM and GRM and eliminate the 19 individuals with genotype-pedigree conflicts, except that 19 rows and corresponding columns were deleted. In the standard $\mathbf{G}$, the additive genetic variance attributed to each SNP genotype is equally important and GRM are identical for all traits. Recently, methodologies for constructing $\mathbf{G}$ with weighting factors to account for locus-specific variances were proposed (WANG, H et al., 2012; Su et al., 2014; Calus et al., 2014). The method reported by Wang et al. (WANG, H et al., 2012) calculates SNP effects from the solution of SSGBLUP-I and then reconstructs a new GRM using weights that are obtained from the previously calculated SNP effects. This can be repeated iteratively to obtain a sequence of GRM. In this approach, GRM will differ for each trait.

The prediction model based on the GRM constructed from one iteration was referred to as

SSGBLUP-II and the GRM constructed from five iterations was referred to as SSGBLUP-III. To remove singularity, GRM can be blended with NRM (Aguilar et al., 2010) but this was not done in our study, nor were residual polygenic effects separately modeled in either SSGBLUP or SSBR. Instead, diagonal and off-diagonal elements of $\mathbf{G}$ were separately scaled so that their means equal the corresponding means of $\mathbf{A}_{\text{gg}}$, which is expected to remove the singularity of GRM in SSGBLUP that is introduced by centering the SNPs.

6.3.2.4 Single-step Bayesian regression methods

In the single-step Bayesian regression analyses, the adjusted phenotypes were modeled as:

$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{ZM}\alpha + \mathbf{Z_n}\epsilon + \mathbf{e}$,

where $\mathbf{y}$ is the adjusted phenotypic vector for both genotyped and non-genotyped individuals, $\mathbf{X} = \begin{bmatrix} \mathbf{1} & -\mathbf{Z}_n\mathbf{A}_{\text{ng}}\mathbf{A}_{\text{gg}}^{-1}\,\mathbf{1} \\ \mathbf{1} & -\mathbf{Z}_g\mathbf{1} \end{bmatrix}$, $\boldsymbol{\beta} = \begin{bmatrix} \mu \\ \mu_g \end{bmatrix}$, μ is the overall mean, and μ_g represents the difference in breeding values between genotyped and non-genotyped animals, $\mathbf{Z}$ is the design matrix, $\mathbf{M} = \begin{bmatrix} \hat{\mathbf{M}}_n \\ \mathbf{M}_g \end{bmatrix}$, where $\mathbf{M}_g$ is the matrix of SNP covariates for genotyped animals and $\hat{\mathbf{M}}_n$=$\mathbf{A}_{\text{ng}}\mathbf{A}_{\text{gg}}^{-1}\mathbf{M}_g$, representing imputed SNP covariates for non-genotyped animals that are derived from genotyped relatives, ϵ is the imputation residual, $\mathbf{Z}_n$ and $\mathbf{Z}_g$ are the design matrices allocating records to breeding values of non-genotyped animals and genotyped animals. Flat priors were used for μ and μ_g. The prior for e_i is $e_i \overset{iid}{\sim} N\left(0, \sigma_e^2\right)$ with $(\sigma_e^2|\nu_e, S_e^2) \sim \nu_e S_e^2 \chi^2_{\nu_e}$. The prior for ϵ is $\epsilon|\sigma_g^2 \sim\ N(0, (\ \mathbf{A}_{\text{nn}} - \mathbf{A}_{\text{ng}}\mathbf{A}_{\text{gg}}^{-1}\mathbf{A}_{\text{gn}})\sigma_g^2)$ with $(\sigma_g^2|\nu_g, S_g^2) \sim \nu_g S_g^2 \chi^2_{\nu_g}$. The same priors for SNP effects as in BayesB, BayesC and BayesCπ were used in single-step Bayesian regression methods and were referred to as SSBR-B, SSBR-C, or SSBR-Cπ.

The π values in the subsequent analyses for BayesB, BayesC, SSBR-B and SSBR-C were chosen such that they provided the highest accuracies from five-fold cross-validation. Accuracies in BayesB and BayesC were compared using various π values i.e. 0.9999, 0.999, 0.995, 0.99, 0.98 and, then, in steps from 0.95 to 0.6 decreasing by 0.05.

Analyses were performed with GenSel (Habier et al., 2011b) for BayesB, BayesC and BayesCπ methods using only genotyped animals. Estimated breeding values of PBLUP and SSGBLUP were obtained using the software BLUPF90 (Misztal et al., 2002) modified for genomic

analyses (Aguilar et al., 2010). For SSBR methods, JWAS the Julia package for whole-genome analyses (Cheng et al., 2016) was used.

6.3.2.5 Validation

For each validation set, prediction accuracy was calculated as the correlation between the vector of adjusted phenotypes and the vector of estimated breeding values, divided by the square root of trait heritability. Prediction accuracies from these five-fold cross-validation sets were pooled to obtain a single prediction accuracy that was relevant to the method and trait by weighting each of the five validation correlations by the number of individuals in that set. Regressions of adjusted phenotype on estimated breeding value were calculated for all prediction methods.

6.3.2.6 Genome-wide association studies

Genome-wide association studies (GWAS) were performed using the BayesB method with the π value that had given the highest prediction accuracy, in order to describe the genetic architecture for different traits in terms of window variance (Wolc et al., 2014).

6.4 Results

Predictive accuracies for the four traits obtained with BayesB and BayesC for different π values are in Figure 6.1. For BFT, EMA and MAR, predictive accuracies of BayesB and BayesC were similar, but decreased as π increased, and fewer SNPs were assumed to have non-zero effects. For CWT, we observed a different pattern with accuracies increasing as π increased and accuracies of BayesB being always higher than those of BayesC. These two results suggest that CWT is influenced by a few quantitative trait loci (QTL) that explain a large proportion of the genetic variance. The proportions of genetic variance explained by 1-Mb non-overlapping genomic windows are in Figure 6.2, and demonstrate that the QTL for CWT were larger than those for the other traits.

The π values that maximized the cross-validation accuracies in BayesB were 0.95, 0.98, 0.95, and 0.6 for BFT, CWT, EMA and MAR, respectively, and were used in SSBR-B. The

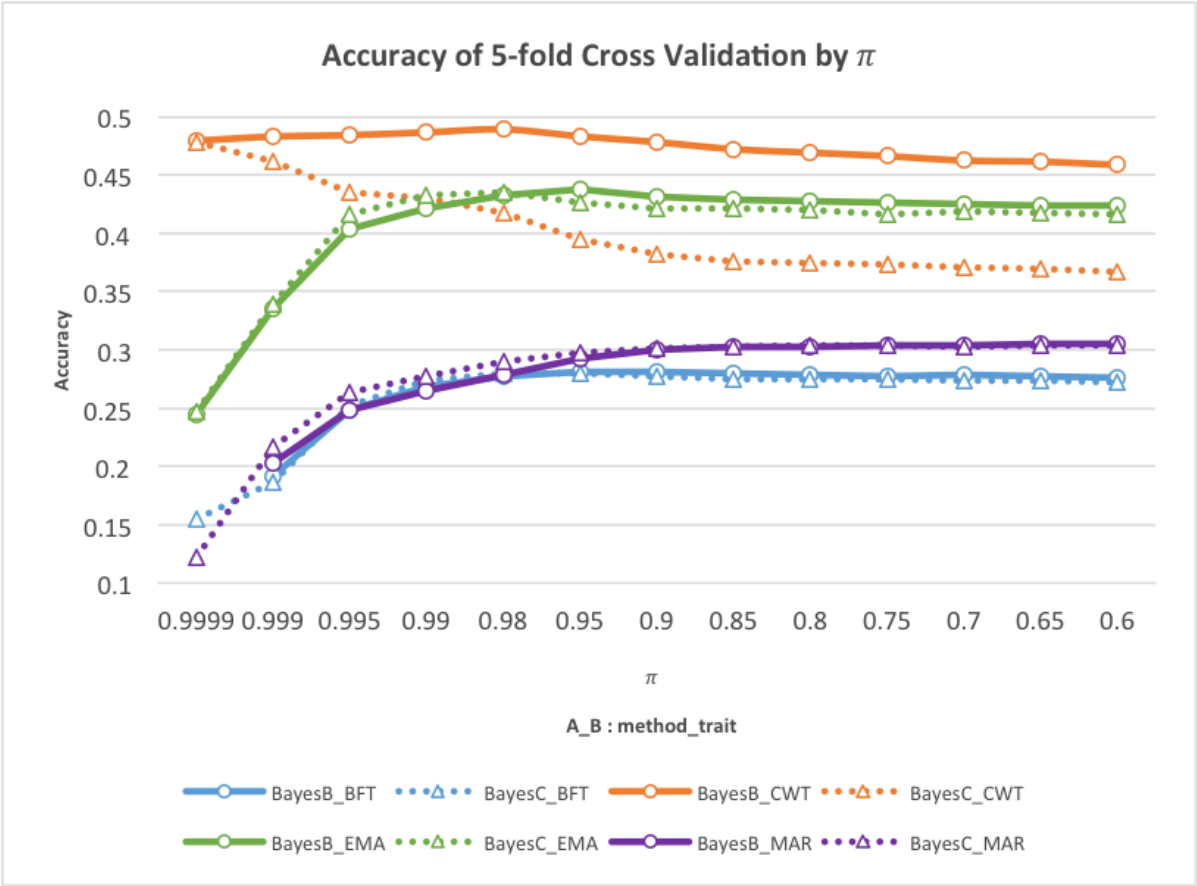

Figure 6.1 Fivefold cross-validation accuracies obtained with BayesB or BayesC using various assumed values for π

corresponding π values in BayesC were 0.98, 0.9999, 0.98, and 0.6 for BFT, CWT, EMA and MAR, respectively, and were used in SSBR-C.

Several windows showed distinctly larger effects than the rest of the genome for BFT and EMA, but the window with the largest effect explained only about 1% of the genetic variance. For MAR, the windows showed smaller effects than those for BFT and EMA with the most significant window explaining less than 0.3% of the genetic variance. These results show that, for BFT and EMA, many QTL each with a small effect are widely distributed across the whole genome, which is consistent with the infinitesimal model. In contrast, for CWT, one window on chromosome 4 and two windows on chromosome 14 explained together more than 15% of the genetic variance while the other windows showed small effects. Using single-SNP association

tests, Lee et al. (Lee et al., 2013) found similar results that indicated that SNPs on chromosome 14 were strongly associated with CWT in Hanwoo beef cattle. These differences in genomic architecture between the four traits probably explain the difference in the pattern of prediction accuracy between CWT and the three other traits as shown in Figure 6.1. BayesB, which shrinks QTL with small effects to a greater extent than BayesC, may capture QTL with large effects better and therefore yield higher prediction accuracies (Wolc et al., 2016). BayesB and BayesC methods with a high π value tend to capture the same few QTL with large effects, thus their similar prediction accuracies.

Prediction accuracies of models SSGBLUP-I and SSBR-C ($\pi = 0$) without estimated variances were identical and equal to 0.351 for BFT, 0.415 for CWT, 0.413 for EMA and 0.377 for MAR as expected since these models with assumed variance parameters are equivalent in terms of prediction of breeding values (Fernando et al., 2014). In practice, variance components are often treated as unknown and are estimated in a separate analysis, e.g. restricted maximum likelihood (REML) followed by GBLUP, or jointly with an informative prior, e.g. BayesB, SSBR-B, etc. The variances of additive genetic effects, SNP effects and residual effects were estimated in the subsequent analyses described below.

To compare methods that use all individuals with those that use only genotyped individuals, prediction accuracies (Figure 6.3) were calculated using PBLUP (all animals) and PBLUP-G (PBLUP using only phenotypes on genotyped animals), BayesB, BayesC, BayesC (π= 0), and BayesCπ, SSGBLUP-I and SSGBLUP-II and SSBR-B, SSBR-C, SSBR-C (π= 0), and SSBR-Cπ.

6.4.0.1 Genomic methods versus pedigree-based BLUP

Methods using genomic information always outperformed PBLUP with the same phenotypic data. Using data from only genotyped animals, accuracies were higher with BayesB, BayesC and BayesCπ than with PBLUP-G for all traits. When data from both genotyped and non-genotyped individuals were used, prediction accuracies of the single-step methods were higher than those of PBLUP for all traits.

6.4.0.2 Single-step methods versus BayesB, BayesC and BayesCπ

For BFT and MAR, prediction accuracies of the single-step methods were higher than those of BayesB, BayesC and BayesCπ. Gains in accuracy with the single-step methods ranged from +0.06 to +0.09 for BFT and from +0.05 to +0.07 for MAR, whereas for EMA, there was no advantage and only a slight gain in accuracy was observed in PBLUP versus PBLUP-G. For CWT, SSBR always outperformed the corresponding Bayesian methods using only genotyped individuals and the gains in accuracy were +0.05 (SSBR-C ($\pi = 0$) versus BayesC ($\pi = 0$)), +0.01 (SSBR-C ($\pi = 0.9999$) versus BayesC ($\pi = 0.9999$)), +0.10 (SSBR-Cπ versus BayesCπ) and +0.04 (SSBR-B ($\pi = 0.98$) versus BayesB ($\pi = 0.98$)). However, although information from non-genotyped individuals was incorporated, for CWT prediction accuracy of SSGBLUP was lower than that of BayesC ($\pi = 0.9999$) and BayesB ($\pi = 0.98$) due to the benefits of mixture priors of the SNP effects for this particular trait.

6.4.0.3 Comparisons between single-step methods

The differences in accuracies between single-step methods (yellow and blue bars in Figure 6.3) were small for BFT, EMA and MAR, and a similar pattern was found between Bayesian methods (red bars in Figure 6.3) using only genotyped individuals. For the CWT trait for which the GWAS detected a small number of regions with large effects, prediction accuracies differed with the method used. With the benefits of mixture priors and information from non-genotyped individuals, prediction accuracies of the SSBR methods, especially SSBR-B, were higher (+0.09) than those of the SSGBLUP methods. As for the SSBR methods with mixture priors, the SSGBLUP methods, which use weighted GRM (SSGBLUP-II and SSGBLUP-III), showed higher accuracies than SSGBLUP-I for CWT. Prediction accuracy of SSGBLUP-II was similar to that of SSGBLUP-I for EMA and MAR but lower for BFT. Prediction accuracy of SSGBLUP-III was lower than that of SSGBLUP-I for EMA, MAR and BFT. Regressions of adjusted phenotype on estimated breeding value did not show large differences among methods, but SSGBLUP-II and SSGBLUP-III had the lowest coefficients for all traits, much lower than 1, which indicates that their genomic predictions are biased upwards (Table 6.1).

Table 6.1 Regression coefficient of adjusted phenotype on estimated breeding values for backfat (BFT), carcass weight (CWT), eye-muscle area (EMA) and marbling (MAR) traits

	Trait			
	BFT	**CWT**	**EMA**	**MAR**
SSBR-C(π estimation)	0.85	0.97	0.99	0.88
SSBR-B(π =chosen[a])	0.88	1.08	1.07	0.74
SSBR-C(π =chosen[b])	0.88	1.02	1.04	0.89
SSBR-C(π =0)	0.86	1.21	1.00	0.87
BayesC(π estimation)	0.82	1.05	1.05	0.86
BayesB(π =chosen[a])	0.82	1.03	1.26	0.70
BayesC(π =chosen[b])	0.88	1.06	1.12	0.87
BayesC(π =0)	0.86	1.20	1.09	0.88
SSGBLUP-I	0.73	1.15	0.97	0.79
SSGBLUP-II	0.54	0.84	0.75	0.64
SSGBLUP-III	0.52	0.90	0.79	0.61
PBLUP	0.76	1.12	1.02	0.93
PBLUP-G	0.61	1.33	1.30	0.92

6.5 Discussion

Prediction accuracies of all methods using genomic information were higher than those of pedigree-based BLUP. However, the degree of superiority of genomic selection differed between methods and traits.

Simultaneous use of all pedigree, phenotypic and genomic information in single-step methods improved prediction accuracy relative to methods that only use data from genotyped animals for all traits, except EMA. For EMA, there was little benefit from including the extra data in the PBLUP analyses (compared to PBLUP-G). Although it is not certain why the additional phenotypes from non-genotyped individuals resulted in no real gain in accuracy for EMA, we hypothesize that the superiority of PBLUP over PBLUP-G is related to the gain in accuracy that can be expected by SSBR-C relative to BayesC.

Both SSBR and SSGBLUP methods showed similar prediction accuracies when the genetic architecture appeared to approach the infinitesimal model as was the case for BFT, EMA, and MAR. However, for CWT, prediction accuracies of the SSBR methods were higher than those

of SSGBLUP when there were only a few QTL with large effects. For that trait, the SSBR methods benefited from the use of the mixture priors.

The largest benefit of the SSBR methods was reached when an appropriate π was applied. However, it is computationally intensive to find this value of π through cross-validation. Methods for estimating π are beneficial, but they require large data sets. An appropriate π was more critical for the Bayesian methods that only used genotyped individuals than for the SSBR methods. For example, differences in prediction accuracies between BayesC ($\pi = 0.9999$) and BayesCπ reached values of 0.10 but only of 0.01 between SSBR-C ($\pi = 0.9999$) and SSBR-Cπ. Presumably, priors become less important in the single-step analyses where more data are used.

Three factors can result in increased accuracy. First, the inclusion of genomic information, which was revealed when genomic methods were compared to pedigree-based BLUP. Second, the use of additional phenotypic information from including non-genotyped individuals, which was shown by comparing Bayesian methods using only genotyped animals with their corresponding single-step methods. Third, the use of methods that exploit genomic regions with large effects, as was found for one of the four traits using either mixture priors or iterative weighted methods for computing GRM.

SSGBLUP with iterative calculation of weighted genomic matrices had the disadvantage that it reduced prediction accuracy and increased bias for traits that were not associated with genomic regions with large effects, whereas the Bayesian models with mixture priors performed comparably regardless of the genomic architecture. SSGBLUP with iterative calculation of weighted genomic matrices shrinks small effects to zero, and more so with each additional iteration. There is no statistical basis to determine the optimal number of iterations except by trial and error, and neither one nor five iterations resulted in improvements in this dataset.

In this study, which is based on a small population of Hanwoo cattle, prediction accuracy was higher for all genomic evaluations compared to pedigree-based BLUP. In such a situation, where the genomic reference population is relatively small, single-step methods, which can routinely account for genomic regions with large effects when they are present, are recommended for additional gains in accuracy.

The "single-step" methodologies, which take advantage of all pedigree, phenotypic and

genomic information simultaneously, give similar or higher prediction accuracies compared to methods using only genotyped individuals. Compared to SSGBLUP, the SSBR methods showed additional benefit for the CWT trait, which is associated with QTL with large effects. There is no disadvantage in using SSBR methods for all traits.

6.6 Authors contributions

JL and HC conceived the study, undertook the analysis and wrote the draft. DG, RF, JD, BG contributed to the analysis. DG, RF contributed to the final version of manuscript. All authors read and approved the final manuscript.

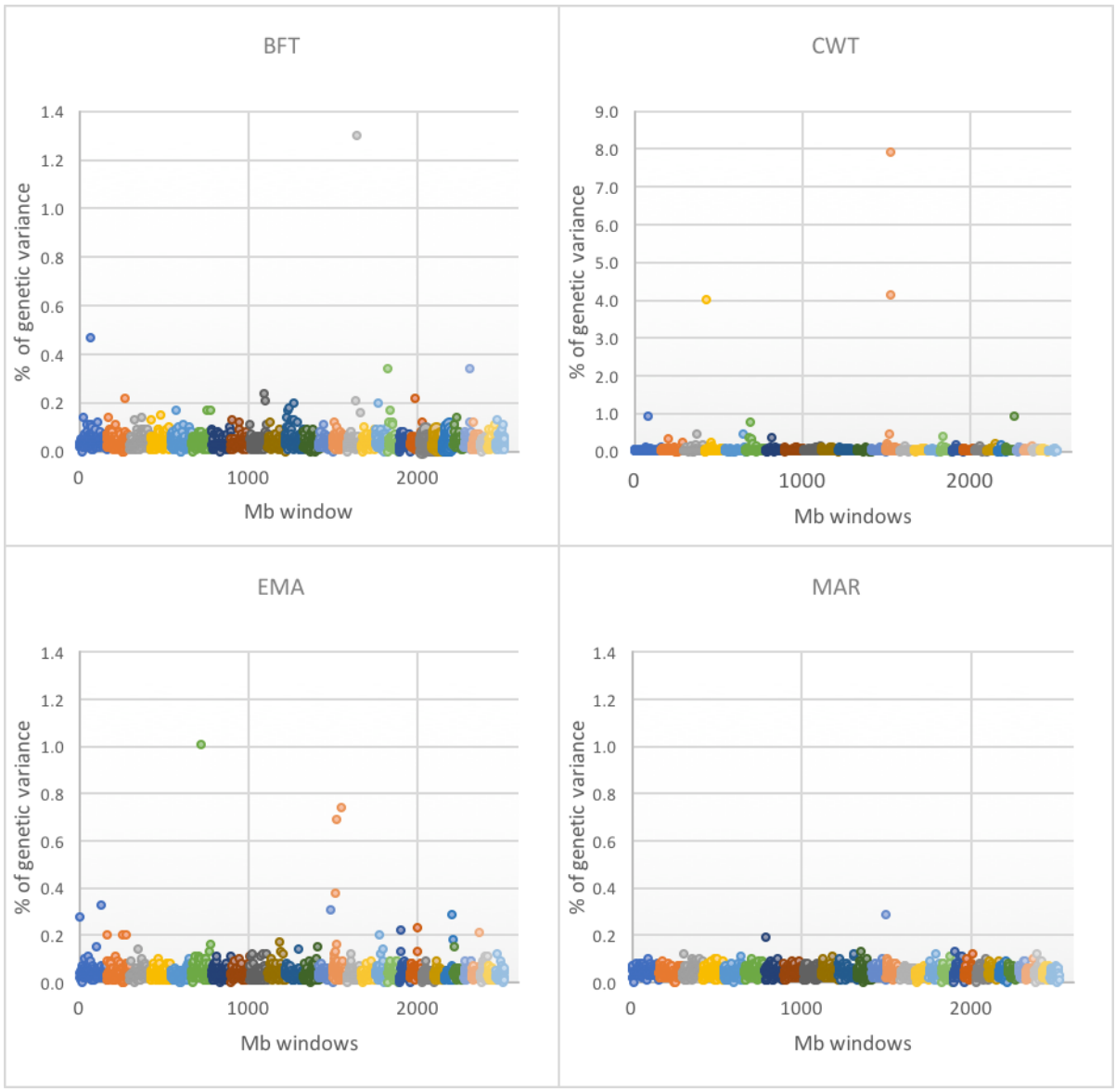

Figure 6.2 Results of the GWAS for each of the four traits. Different colors represent different autosomes (ordered from 1 to 29)

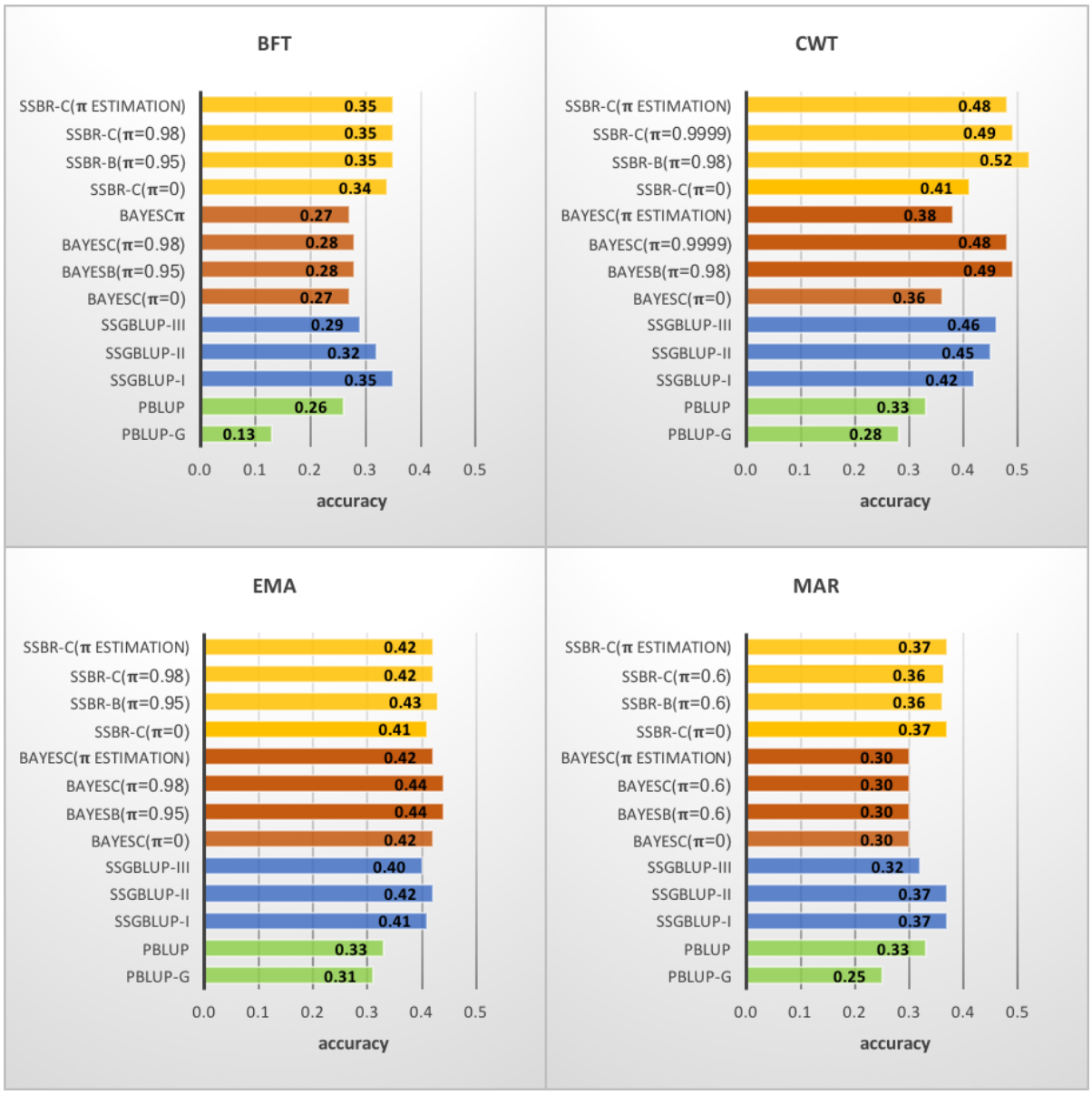

Figure 6.3 Prediction accuracies by cross-validation for a variety of methods applied to backfat (BFT), carcass weight (CWT), eye-muscle area (EMA) and marbling (MAR). Conventional PBLUP based on only genotyped individuals (PBLUP-G) or using all animals (PBLUP), BayesB with chosen π (BAYESC(π = chosen value)), BayesC with chosen (BAYESC (= chosen value)) BayesC with = 0 (BAYESC ($\pi = 0$)) or BayesC estimating π (BAYESC (π ESTIMATION)), single-step genomic BLUP constructing two different genomic relationship matrix (SSGBLUP-I and SSGBLUP-II) and single-step Bayesian regression corresponding to Bayesian methods (SSBR-B (π = chosen value), SSBR-C (π = chosen value), SSBR-C ($\pi = 0$), and SSBR-C (π ESTIMATION)).

CHAPTER 7. EFFICIENT STRATEGIES FOR LEAVE-ONE-OUT CROSS VALIDATION FOR GENOMIC BEST LINEAR UNBIASED PREDICTION

Hao Cheng, Dorian Garrick and Rohan Fernando

A paper published in Journal of Animal Science and Biotechnology

7.1 Abstract

A random multiple-regression model that simultaneously fit all allele substitution effects for additive markers or haplotypes as uncorrelated random effects was proposed for Best Linear Unbiased Prediction, using whole-genome data. Leave-one-out cross validation can be used to quantify the predictive ability of a statistical model. Naive application of Leave-one-out cross validation is computationally intensive because the training and validation analyses need to be repeated n times, once for each observation. Efficient Leave-one-out cross validation strategies are presented here, requiring little more effort than a single analysis.

Efficient Leave-one-out cross validation strategies is 786 times faster than the naive application for a simulated dataset with 1000 observations and 10,000 markers and 99 times faster with 1000 observations and 100 markers. These efficiencies relative to the naive approach using the same model will increase with increases in the number of observations.

Efficient Leave-one-out cross validation strategies are presented here, requiring little more effort than a single analysis.

7.2 Introduction

A random multiple-regression model that simultaneously fit all allele substitution effects for additive markers or haplotypes as uncorrelated random effects was proposed for Best Linear Unbiased Prediction (BLUP) (Meuwissen et al., 2001a), using whole-genome data. Breeding values are defined as the sum of the effects of all the markers or haplotypes, and their estimates are widely used for prediction of the merit of selection candidates. Estimates of marker or haplotype effects are used to predict breeding values of individuals that were not present in a previous analysis commonly referred to as training. An alternative earlier published approach to use marker or haplotype information fits breeding values as random effects based on covariances defined by a "genomic relationship matrix" computed from genotypes (Nejati-Javaremi et al., 1997). These two models have been shown to be equivalent in terms of predicting breeding values (Fernando, 1998; Strandén and Garrick, 2009) and we refer to them here as marker effect models (MEM) or breeding value models (BVM), the latter often known as Genomic Best Linear Unbiased Prediction (GBLUP).

Cross validation is often used to quantify the predictive ability of a statistical model. In k-fold cross validation, the whole dataset is partitioned into k parts with k analyses, where one part is omitted for training with validation on the omitted part. Leave-one-out cross validation (LOOCV) is a special case of k-fold cross validation with $k = n$, the number of observations. When the dataset is small, leave-one-out cross validation is appealing as the size of the training set is maximized. However, naive application of LOOCV is computationally intensive, requiring n analyses.

We show below how LOOCV can be performed using either the MEM or BVM with little more effort than is required for a single analysis with n observations.

7.3 Materials and Methods

Use of the MEM is more efficient when the number n of individuals is larger than the number p of markers, because for this model the mixed model equations are of order p plus the number of other effects. When $n < p$, estimated breeding values can be obtained more efficiently by

solving the mixed model equations for the BVM of order n plus the number of other effects. We deal with the special case where the only other effect is a general mean and phenotypes have been pre-corrected for other nuisance variables. Efficient strategies for LOOCV using this special case for MEM when $n \geq p$ and BVM when $p \geq n$ are shown below.

7.3.1 Marker effect models

The MEM for GBLUP can be written as

$$\boldsymbol{y} = \mathbf{1}\mu + \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{e}, \tag{7.1}$$

where $\boldsymbol{y}$, a $n \times 1$ vector for phenotypes, has been pre-corrected for all fixed effects other than μ, the overall mean, $\boldsymbol{X}$ is the $n \times p$ matrix of marker covariates, $\boldsymbol{\beta}$ is a $p \times 1$ random vector of the allele substitution effects and $\boldsymbol{e}$ is a $n \times 1$ random vector of residuals. Often it is assumed that marker effects are identically and independently distributed (iid) random variables with null means and variances σ^2_β. Thus, under the usual assumption that the residuals are iid with null means and variances σ^2_e, $E(\mathbf{y}) = \mathbf{1}\mu$. When MEM is used, LOOCV can be performed by using a well-known strategy used in least-squares regression to compute the predicted residual sum of square (PRESS) (Allen, 1974) statistic.

7.3.1.1 LOOCV strategy for MEM

BLUP of $\boldsymbol{\beta}^* = \begin{bmatrix} \mu \\ \boldsymbol{\beta} \end{bmatrix}$ can be obtained by solving the mixed model equations

$$\left(\boldsymbol{X}^{*\prime}\boldsymbol{X}^* + \boldsymbol{D}\lambda\right)\hat{\boldsymbol{\beta}}^* = \boldsymbol{X}^{*\prime}\boldsymbol{y}, \tag{7.2}$$

where $\boldsymbol{X}^* = \begin{bmatrix} \mathbf{1} & \boldsymbol{X} \end{bmatrix}$, $\hat{\boldsymbol{\beta}}^* = \begin{bmatrix} \hat{\mu} \\ \hat{\boldsymbol{\beta}} \end{bmatrix}$, $\boldsymbol{D}$ is a diagonal matrix whose elements are 0 followed by a p vector of 1s and $\lambda = \frac{\sigma^2_e}{\sigma^2_\beta}$.

Now, BLUP for $\boldsymbol{\beta}^*_{-\boldsymbol{j}}$, where observation j is left out, can be obtained as

$$\hat{\boldsymbol{\beta}}^*_{-\boldsymbol{j}} = \left(\boldsymbol{X}^{*\prime}_{-j}\boldsymbol{X}^*_{-j} + \boldsymbol{D}\lambda\right)^{-1}\boldsymbol{X}^{*\prime}_{-j}\boldsymbol{y}_{-j}, \tag{7.3}$$

where $\boldsymbol{X}^*_{-j}$ is $\boldsymbol{X}^*$ with the jth row removed and $\boldsymbol{y}_{-j}$ is $\boldsymbol{y}$ with the jth element removed.

Suppose $\boldsymbol{x^*}'_j$ is the jth row of $\boldsymbol{X^*}$, then from the matrix inverse lemma (Strandén and Garrick, 2009) ,

$$\begin{aligned}\left(\boldsymbol{X^*}'_{-j}\boldsymbol{X^*}_{-j}+\boldsymbol{D}\lambda\right)^{-1} &= \left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda-\boldsymbol{x^*}_j\boldsymbol{x^*}'_j\right)^{-1}\\ &= \left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}-\frac{\left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}\boldsymbol{x^*}_j\boldsymbol{x^*}'_j\left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}}{1-H_{jj}},\end{aligned} \tag{7.4}$$

where the quadratic $H_{jj}=\boldsymbol{x^*}'_j\left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}\boldsymbol{x^*}_j$ is the jth diagonal element of $\boldsymbol{H}=\boldsymbol{X^*}\left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}\boldsymbol{X^{*\prime}}$.

Using (7.3) in (7.4), the prediction residual for the jth observation can be written as

$$\begin{aligned}\hat{e_j} &= y_j-\boldsymbol{x^*}'_j\hat{\boldsymbol{\beta}}^*{}_{-j}\\ &= y_j-\boldsymbol{x^*}'_j\left[\left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}-\frac{\left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}\boldsymbol{x^*}_j\boldsymbol{x^*}'_j\left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}}{1-H_{jj}}\right]\boldsymbol{X^{*\prime}}_{-j}\boldsymbol{y}_{-j}\\ &= \frac{\left(1-\boldsymbol{H}_{jj}\right)y_j-\boldsymbol{x^*}'_j\left(\boldsymbol{X^{*\prime}X^*}+\boldsymbol{D}\lambda\right)^{-1}\boldsymbol{X^{*\prime}}_{-j}\boldsymbol{y}_{-j}}{1-\boldsymbol{H}_{jj}} \qquad (7.5)\\ &= \frac{y_j-\boldsymbol{x^*}'_j\hat{\boldsymbol{\beta}}^*}{1-\boldsymbol{H}_{jj}}. \qquad (7.6)\end{aligned}$$

These prediction errors can be squared and accumulated over n realizations to compute PRESS defined as $\sum_{j=1}^{n}\hat{e_j}^2$. The accuracy of genomic prediction is often quantified as the correlation between the predicted and observed values of y_j, and that correlation can be estimated from the values of $\hat{y}_j$, which can be computed efficiently as $\hat{y}_j=y_j-\hat{e}_j$, using the observed values of y_j. When a specific group of individuals is of interest, prediction accuracies and PRESS can also be calculate using $\hat{e_j}$ for individuals in that group.

7.3.2 Breeding value models

When $n<p$, the genomic prediction of the breeding value $\mathbf{x}'_j\hat{\boldsymbol{\beta}}$ can be obtained more efficiently by solving the mixed model equations for the BVM:

$$\boldsymbol{y}=\mathbf{1}\mu+\mathbf{Zu}+\boldsymbol{e}, \tag{7.7}$$

where $\mathbf{u}=\mathbf{X}\beta$, $\text{var}(\mathbf{u})=\mathbf{XX}'\sigma^2_\beta$, $\mathbf{Z}$ is the identity matrix of order n and other variables are as in the MEM. Further, in both models $\text{E}(\mathbf{y})=\mathbf{1}\mu$, and $\text{var}(\mathbf{y})=\mathbf{XX}'\sigma^2_\beta+\mathbf{I}\sigma^2_e$. These two

models are said to be equivalent (Henderson, 1984), and linear functions predicted from one model are identical to corresponding predictions from the other model. Two efficient strategies for LOOCV using the BVM are shown below.

7.3.2.1 LOOCV strategy I for BVM

The mixed model equations for this model are:

$$\left(\boldsymbol{Z^{*'}Z^{*}} + \boldsymbol{G}\lambda\right)\hat{\boldsymbol{u}}^{*} = \boldsymbol{Z^{*'}y}, \tag{7.8}$$

where $\boldsymbol{Z^{*}} = \begin{bmatrix}\mathbf{1} & \boldsymbol{Z}\end{bmatrix}$, $\hat{\boldsymbol{u}}^{*} = \begin{bmatrix}\hat{\mu}\\ \hat{\boldsymbol{u}}\end{bmatrix}$, $\mathbf{G} = \begin{bmatrix}0 & \mathbf{0}'\\ \mathbf{0} & (\boldsymbol{XX'})^{-1}\end{bmatrix}$. Due to the relative order of the coefficient matrices for the MEM and the BVM, when $n < p$, $\boldsymbol{x^{*'}}_{j}\hat{\boldsymbol{\beta}}^{*}$ is more efficiently obtained as $\hat{u^{*}}_{j}$. Similarly, $\text{var}(\boldsymbol{x^{*'}}_{j}\boldsymbol{\beta}^{*} - \boldsymbol{x^{*'}}_{j}\hat{\boldsymbol{\beta}}^{*}) = \mathbf{x^{*'}}_{j}\left(\mathbf{X^{*'}X^{*}} + \mathbf{D}\lambda\right)^{-1}\mathbf{x^{*}}_{j}\sigma_{e}^{2}$ can be obtained more efficiently as $\text{var}(u_{j}^{*} - \hat{u^{*}}_{j}) = \mathbf{z^{*'}}_{j}\left(\mathbf{Z^{*'}Z^{*}} + \mathbf{G}\lambda\right)^{-1}\mathbf{z^{*}}_{j}\sigma_{e}^{2}$. Using these two equalities, the formula for $\hat{e}_{j}$ becomes:

$$\begin{aligned}
\hat{e_{j}} &= y_{j} - \boldsymbol{x^{*'}}_{j}\hat{\boldsymbol{\beta}}^{*}{}_{-j}\\
&= \frac{y_{j} - \boldsymbol{x^{*'}}_{j}\hat{\boldsymbol{\beta}}^{*}}{1 - H_{jj}}\\
&= \frac{y_{j} - \boldsymbol{x^{*'}}_{j}\hat{\boldsymbol{\beta}}^{*}}{1 - \boldsymbol{x^{*'}}_{j}\left(\mathbf{X^{*'}X^{*}} + \mathbf{D}\lambda\right)^{-1}\mathbf{x^{*}}_{j}}\\
&= \frac{y_{j} - \mathbf{z^{*'}}_{j}\hat{u^{*}}}{1 - \mathbf{z^{*'}}_{j}\left(\mathbf{Z^{*'}Z^{*}} + \mathbf{G}\lambda\right)^{-1}\mathbf{z^{*}}_{j}}\\
&= \frac{y_{j} - \mathbf{z^{*'}}_{j}\hat{u^{*}}}{1 - \boldsymbol{C}_{jj}},
\end{aligned} \tag{7.9}$$

where the quadratic $\mathbf{z^{*'}}_{j}\left(\mathbf{Z^{*'}Z^{*}} + \mathbf{G}\lambda\right)^{-1}\mathbf{z^{*}}_{j}$ is the jth diagonal element of $\boldsymbol{C} = \boldsymbol{Z^{*}}\left(\boldsymbol{Z^{*'}Z^{*}} + \boldsymbol{G}\lambda\right)^{-1}\boldsymbol{Z^{*'}}$.

7.3.2.2 LOOCV strategy II for BVM

Another efficient strategy for BVM is shown here. First we consider the situation where $\boldsymbol{y}$ has been pre-corrected for μ in addition to nuisance effects so that $\text{E}(\mathbf{y}) = \mathbf{0}$ and we define $\mathbf{var}\,(\boldsymbol{y}) = \mathbf{XX}'\sigma_{\beta}^{2} + \mathbf{I}\sigma_{e}^{2} = \boldsymbol{V}$. Now matrix $\mathbf{Q}$ is constructed by augmenting the covariance

matrix of $\boldsymbol{y}$ with one leading row and column as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{y}'\mathbf{y} & \mathbf{y}^{'} \\ \mathbf{y} & \mathbf{V} \end{bmatrix}.$$

To obtain the prediction error for observation j, the second row and column of $\mathbf{Q}$ are permuted with row and column $j+1$. In this manner $\mathbf{Q}$ has its rows and columns symmetrically permuted as $\mathbf{P}^{'}{}_j\mathbf{Q}\mathbf{P}_j = \boldsymbol{W}$, where the permutation matrix $\mathbf{P}_j$ is obtained by permuting the second row of the n order identity matrix with row $j+1$. So the permuted matrix is:

$$\mathbf{W} = \begin{bmatrix} \mathbf{y}'\mathbf{y} & y_j & \mathbf{y}^{'}_{-j} \\ y_j & V_{jj} & \mathbf{V}_{j,-j} \\ \mathbf{y}_{-j} & \mathbf{V}_{-j,j} & \mathbf{V}_{-j,-j} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^{'} & \mathbf{C} \end{bmatrix}$$

where we will define the leading 2×2 matrix as $\mathbf{A} = \begin{bmatrix} \mathbf{y}'\mathbf{y} & y_j \\ y_j & V_{jj} \end{bmatrix}$, and the other partitions as $\mathbf{B} = \begin{bmatrix} \mathbf{y}^{'}_{-j} \\ \mathbf{V}_{j,-j} \end{bmatrix}$, and $\mathbf{C} = \mathbf{V}_{-j,-j}$, where $-j$ denotes that the jth element, row or column has been removed. Defining $\mathbf{W^{11}}$ as the top left or leading 2×2 sub-matrix in $\mathbf{W^{-1}}$ corresponding to the position of $\mathbf{A}$ in $\mathbf{W}$, and using partitioned inverse-matrix identities (Searle, 1982), the inverse of $\mathbf{W^{11}}$ can be written as,

$$\begin{aligned} (\mathbf{W^{11}})^{-1} &= \mathbf{A} - \mathbf{B}\mathbf{C^{-1}}\mathbf{B}^{'} \\ &= \begin{bmatrix} \boldsymbol{y'y} & y_j \\ y_j & V_{jj} \end{bmatrix} - \begin{bmatrix} \boldsymbol{y}^{'}_{\mathbf{-j}} \\ \mathbf{V_{j,-j}} \end{bmatrix} \mathbf{V}^{-1}_{-j,-j} \begin{bmatrix} \boldsymbol{y}_{\mathbf{-j}} & \mathbf{V_{-j,j}} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{y'y} - \boldsymbol{y}^{'}_{-j}\mathbf{V^{-1}_{-j,-j}}\boldsymbol{y}_{-j} & y_j - \boldsymbol{y}^{'}_{-j}\mathbf{V}^{-1}_{-j,-j}\boldsymbol{V}_{-j,j} \\ y_j - \mathbf{V_{j,-j}}\mathbf{V^{-1}_{-j,-j}}\boldsymbol{y}_{-j} & V_{jj} - \mathbf{V_{j,-j}}\mathbf{V^{-1}_{-j,-j}}\mathbf{V_{-j,j}} \end{bmatrix}. \end{aligned} \tag{7.10}$$

Now $\boldsymbol{V_{j,-j}}$ in element $(2,1)$ of the above inverse matrix is the vector of covariances between y_j and $\mathbf{y}_{-j}$ and $\mathbf{V^{-1}_{-j,-j}}$ is the inverse of the covariance matrix of $\mathbf{y}_{-j}$. Thus, $\hat{y}_j = \boldsymbol{V}_{j,-j}\boldsymbol{V}^{-1}_{-j,-j}\mathbf{y}_{-j}$ is the Best Linear Predictor (BLP) of y_j given $\mathbf{y}_{-j}$, and element (2,1) of (7.10) is the prediction error of y_j. The element $(2,2)$ in (7.10) is the prediction error variance (PEV) for y_j, where $PEV = var(y_j - \hat{y}_j)$. PEV can also be used to calculate theoretical

reliability for individual i as $1 - \frac{PEV_i}{V_{jj}}$, and characterizing the distributions of reliability for all the individuals in a dataset has a number of practical applications. Note this allows us to obtain the PEV of every individual and the distribution of these values provide information as to the robustness of genomic predictions across the population of individuals represented in the dataset. This PEV is determined by the genomic variance-covariance matrix and does not depend on $\mathbf{y}$. Two different datasets could generate the same PRESS statistic but with different distributions of PEV.

Now, because the permutation matrix $\mathbf{P}_j$ is orthogonal, $\mathbf{W}^{-1} = (\mathbf{P}_j'\mathbf{Q}\mathbf{P}_j)^{-1} = \mathbf{P}_j'\mathbf{Q}^{-1}\mathbf{P}_j$, and the elements of $\mathbf{W}^{11}$ that are of interest in terms of predicting individual j can be obtained directly from $\mathbf{Q}^{-1}$ as

$$\mathbf{W}^{11} = \begin{bmatrix} q^{1,1} & q^{1,(1+j)} \\ q^{(1+j),1} & q^{(1+j),(1+j)} \end{bmatrix}. \tag{7.11}$$

It follows that $\hat{e}_j$, which is the off-diagonal element of the inverse of the 2×2 matrix $\mathbf{W}^{11}$, can be written in terms of $\mathbf{Q}^{-1}$ as

$$\hat{e_j} = \frac{-q^{(1+j),1}}{q^{1,1}q^{(1+j),(1+j)} - q^{1,(1+j)}q^{(1+j),1}}, \tag{7.12}$$

where $q^{i,j}$ is the element from row i and column j of $\mathbf{Q}^{-1}$. Thus, once $\mathbf{Q}^{-1}$ is computed, $\hat{e}_j$ for all j can be computed using (7.12), and these values can be used to compute PRESS as $\sum_{j=1}^{n} \hat{e_j}^2$. To estimate the correlation between the predicted and observed values of y_j, the value of $\hat{y}_j$ is efficiently obtained as the difference $\hat{y}_j = y_j - \hat{e}_j$.

Now we consider the situation without pre-correcting $\mathbf{y}$ for μ, where $\mathrm{E}(\mathbf{y}) = \mathbf{1}\mu$. Now the mixed model (7.7) contains both fixed and random effects. Note that the mixed model equations that correspond to this mixed effects model can be derived by treating μ as "random" with null mean and large variance. So, let

$$var\,(\boldsymbol{\beta}^*) = \begin{bmatrix} \sigma_L^2 & \mathbf{0}' \\ \mathbf{0} & I\sigma_\beta^2 \end{bmatrix} = \boldsymbol{\Sigma},$$

for sufficiently large value of σ_L^2. Then under this assumption, $\mathrm{E}(\mathbf{y}) = \mathbf{0}$ and $var\,(\boldsymbol{y}) = \mathbf{X}^*\boldsymbol{\Sigma}\boldsymbol{X}^{*'} + \mathbf{I}\sigma_e^2 = \boldsymbol{V}^*$, and thus $\hat{y}_j = \boldsymbol{V}^*_{j,-j}\boldsymbol{V}^{*-1}_{-j,-j}\mathbf{y}_{-j}$ is the BLP from the random effects

rather than mixed effects model of y_j given $\mathbf{y}_{-j}$. This BLP obtained from the model with random μ will be numerically very close to the BLUP obtained from the mixed model with fixed μ. The Q matrix corresponding to the BLP with random μ is constructed as $\begin{bmatrix} \mathbf{y}'\mathbf{y} & \mathbf{y}' \\ \mathbf{y} & \mathbf{V}^* \end{bmatrix}$ and prediction residuals are obtained as (7.12).

7.3.3 Numerical Example

Phenotypes $\boldsymbol{y}$ and genotypes $\boldsymbol{X}$ at 5 markers for 3 individuals are in Table 7.1. Assume $\sigma^2_\beta = \frac{\sigma^2_e}{10}$ and the overall mean μ is the only fixed effect. In LOOCV strategy for MEM and strategy I for BVM, the diagonal elements of $\boldsymbol{H}$ for MEM and $\boldsymbol{C}$ for BVM, which are in the denominators of (7.6) and (7.9), are in Table 7.2. The numerators of (7.6) and (7.9) are obtained by solving the MME (7.2) and (7.8). Then prediction errors are calculated as in (7.6) and (7.9) and shown in Table 7.4. In LOOCV strategy II for BVM, the $\boldsymbol{Q}$ matrix (Table 7.3) is constructed using $\sigma^2_L = 1000$, which is sufficiently large relative to σ^2_e for μ to be indistinguishable from a fixed effect with a flat prior. The prediction errors are calculated as (7.12) and shown in Table 7.4. The MEM strategy and BVM strategy I gave identical prediction errors and identical PRESS for this numerical example were numerically very close to those from the BVM strategy II.

7.3.4 Simulation to compare efficiency

Two datasets were simulated using XSim (Hao Cheng and Fernando, 2015), where 1,000 offspring were sampled from random mating of 100 parents for 10 non-overlapping generations, to compare the computational efficiencies for naive and efficient strategies using BVM or MEM for LOOCV in GBLUP. Dataset I was simulated with 1000 observations and 10,000 SNP markers for a $p \gg n$ scenario. Dataset II was simulated with 1000 observations and 100 markers for a $n \gg p$ scenario. The processor used in the analyses was a 1.4 GHz Intel Core i5 with 4 GB of memory.

All strategies implemented in Julia, a scientific programming language, gave virtually identical prediction accuracies defined as the correlation between $\mathbf{y}$ and $\hat{\mathbf{y}}$ for each dataset. For

dataset I, efficient BVM is 786 times faster than the naive application (3.107s versus 2442.59s) (Table 7.5). For dataset II, efficient MEM is 99 times faster than the naive application (2.979s versus 0.030s) (Table 7.5).

7.4 Discussion

In genomic prediction, the candidates to be predicted are often offspring that are genotyped but not yet phenotyped. In this situation, LOOCV using all individuals in the training dataset will provide an upper bound for the accuracy of prediction, because ancestors in the training dataset with large numbers of descendants have more accurate predictions than descendants. A better estimate of the accuracy of prediction can be obtained by applying LOOCV to only terminal offspring in the training dataset.

Efficient strategies for LOOCV in GBLUP are presented in this paper. LOOCV strategy I and II for BVM are more efficient when $p \gg n$. LOOCV strategy for MEM is more efficient when $n \gg p$. The accuracy of genomic prediction is often quantified as the correlation between the predicted and observed values of y_j, and this correlation can be estimated efficiently using LOOCV strategies. Compared to naive application of LOOCV, which is computationally intensive, LOOCV can be implemented efficiently.

Author's contributions

All authors contributed to the development of the statistical methods. HC wrote the program code and conducted the analyses. The manuscript was prepared by HC and RLF. All authors read and approved the final manuscript.

Table 7.1 phenotypes and genotypes at 5 markers for 3 individuals used in the numerical example

	m1	m2	m3	m4	m5	phenotypes
1	1	2	1	2	2	1.97
2	2	1	0	1	1	2.12
3	0	0	2	1	2	-0.62

Table 7.2 diagonal elements of $\boldsymbol{H}$ in LOOCV strategy for BVM and $\boldsymbol{C}$ for MEM

	$j=1$	$j=2$	$j=3$
H_{jj}	0.46	0.51	0.55
C_{jj}	0.46	0.51	0.55

Table 7.3 Q matrix in strategy II for BVM

	1	2	3	4
1	8.75	1.97	2.12	-0.62
2	1.97	1002.40	1000.80	1000.80
3	2.12	1000.80	1001.70	1000.30
4	-0.62	1000.80	1000.30	1001.90

Table 7.4 prediction errors from different LOOCV strategies (different strategies gave identical prediction errors)

	$j=1$	$j=2$	$j=3$
$\hat{e_j}$	1.13	1.21	-2.66

Table 7.5 Efficiency of alternative LOOCV strategies for GBLUP. Results are given for the computing time in seconds using naive MEM, naive BVM, efficient MEM, efficient BVM I and efficient BVM II.

	Alternative LOOCV Strategies				
	naive MEM	naive BVM	efficient MEM	efficient BVM I	efficient BVM II
$n = 1000$; $p = 10,000$	9490.608	2442.590	105.141	3.107	5.945
$n = 1000$; $p = 100$	2.979	169.928	0.030	2.725	0.217

CHAPTER 8. GENERAL CONCLUSIONS

This thesis considered several statistical models and computational algorithms, which contribute to three areas of research and development in whole genome analyses that include collection or simulation of genomic data, use of genomic data for prediction or GWAS, and validation of the performance of these analyses. The proposed methods improved either the prediction accuracy or the computational efficiency of whole genome analyses. A summary of these methods is in Table 8.1.

In Chapter 2, we proposed a strategy that is efficient in memory usage and computing time to simulate descendants forward in time from ancestors with any density of variant information up to and including sequence data, which can be obtained for founders by sequencing or simulation. This strategy has been implemented in both C++ and Julia versions of XSim.

In Chapter 3, we showed how Gibbs samplers without the Metropolis-Hastings (MH) algorithm can be used for the BayesB method. By introducing a Bernoulli variable δ_j, indicating whether the marker effect for a locus is zero or non-zero, the marker effect and locus-specific variance can be sampled using the Gibbs sampler without use of the MH algorithm. Among the Gibbs samplers that were considered, the blocking Gibbs sampler, where β_j and δ_j were sampled jointly, was the most efficient. This sampler was about 2.1 times as efficient as the MH algorithm proposed by Meuwissen et al. and 1.7 times as efficient as that proposed by Habier et al. In Chapter 4, we proposed a strategy to parallelize Gibbs sampling for each marker within each step of the MCMC chain. This parallelization is accomplished by using an orthogonal data augmentation strategy, where the marker covariate matrix is augmented by adding p new rows such that its columns are orthogonal. The full conditional distributions that are needed for BayesC with orthogonal data augmentation (BayesC-ODA) were derived and the convergence of BayesC-ODA was studied. In analyses of the simulated data, BayesC-ODA

provided virtually identical predictions of breeding values as BayesC when the chain length was about 20,000 to 80,000, which is similar to the commonly used chain length of 50,000. In both Chapters 3 and 4, data augmentation strategies, in spite of introducing more unknowns into the analysis, helped improve the computational efficiency of Bayesian multiple-regression analyses. As expected, the model with augmented data required a longer MCMC chain to get reliable results. However, the computations required for each step in the chain took less time resulting in the speedup of the analyses. Further, the parallel Gibbs sampler also has the advantage of requiring less memory. Both the efficient Gibbs sampler proposed in Chapter 3 and the parallel Gibbs proposed in Chapter 4, resulted in speeding up of whole genome analyses. In addition, use of the parallel Gibbs sampler in Chapter 4 will also reduce the memory requirement for Bayesian multiple-regression analyses.

In Chapter 5, we proposed a flexible variable selection model for multiple-trait analyses with BayesCπ or BayesB priors. A previously proposed multi-trait BayesCπ model (Jia and Jannink, 2012) assumes a locus either affects none of the traits or all of the traits. Our model, however, allows loci to affect any combination of the traits. Our new model was compared to the previously used multi-trait BayesCπ model and single-trait models using real and simulated data. In the real data analyses, multi-trait BayesCπ proposed by Jia et al. and the new model with flexible variable selection provided higher prediction accuracy than single-trait methods and even random regression BLUP, which is equivalent to genomic BLUP. In the simulated data, where a locus had an effect only on one trait, the flexible multi-trait variable selection model had an advantage, when sufficient data were available for the flexible variable selection to be effective. This shows that a more complex prior can be beneficial provided sufficient data to be available. On the other hand, the difference between these methods asymptotically disappeared as the training-set size increased. This shows that differences in priors affect the results only for a range of intermediate values of the training-set size. In Chapter 6, we compared alternative approaches to single-trait genomic prediction using genotyped and non-genotyped Hanwoo beef cattle. In those data analyses, The single-step methods, which take advantage of all pedigree, phenotypic and genomic information simultaneously, gave similar or higher prediction accuracies compared to methods using only genotyped or non-genotyped

individuals. Alternative priors allowed SSBR to outperform SSGBLUP in some cases. Both methods described in Chapter 5 and 6 combined information from other sources of data and improve the prediction accuracy.

In Chapter 7, we proposed efficient LOOCV strategies for GBLUP in scenarios when $n > p$ or $n < p$. These strategies were compared to naive application of LOOCV with simulated data. In these data analyses, efficient LOOCV, requiring little more effort than a single analysis, was much faster than the naive LOOCV.

The performance of Bayesian regression methods proposed in chapter 3-6 were studied in terms of prediction accuracy. As described here, Bayesian regression methods for genomic prediction (Meuwissen et al., 2001a) can also be adapted for GWAS (Fernando et al., 2017). In many GWAS, the association between a single SNP marker and phenotypes is assessed using a mixed model with a fixed effect for that SNP together with a random polygenic effect correlated according to either a pedigree-based or genomic relationship matrix. In these analyses, the association of a single marker with phenotypes is a partial association conditional on all the other markers even when $n > p$. However, most SNPs are highly correlated with neighboring SNPs, and thus, in addition to the neighboring SNPs, the contribution from the SNP fitted as a fixed effect to explain the variability of linked QTL would often be negligible. However, the SNPs in the neighborhood may jointly explain much more of the variability of the QTL. Thus, the association of SNPs in a genomic window, instead of a single SNP, with phenotypes should be assessed in GWAS. Inferences on genomic windows by frequentist method, however, are computationally very intensive, requiring repeated analyses with permutation of the data[]. A benefit of MCMC-based Bayesian regression methods over frequentist methods is that posterior distributions for the proportion of variance attributed to any genomic region can be obtained from a single analysis, using MCMC samples of marker effects . These posterior distributions can be used to make inference on genomic windows based on controlling the posterior type-I error rate to control false positives. One of the advantages of GWAS based on controlling the posterior type-I error rate is that this approach avoids the multiple test penalty that is inherent in using the genome-wise error rate to control false positive, which is typically used in many GWAS.

Methods proposed in chapters 2 and 3 of this thesis to improve the computational efficiency of Bayesian regression methods would contribute to speed up GWAS. Many researchers are interested in pleiotropy and would therefore want to know which loci affect which traits, from a purely biological perspective. Practitioners are often interested in "breaking" the genetic correlation, by selecting parents to give a favorable selection response in respect to multiple trait consequences. In either of these circumstances, with intermediate- rather than asymptotically-large datasets, we believe the multiple-trait Bayesian regression methods proposed in chapter 5 offer real promise. Single-step Bayesian regression methods investigated in chapter 6 provided promise for GWAS to use both genotyped and non-genotyped individuals to reduce selection bias.

Table 8.1 Summary of statistical models and computational algorithms proposed or investigated in the thesis. BayesB, BayesC and BayesCπ are whole-genome Bayesian multiple-regression methods with mixture priors; n is the number of individuals and and p is the number of markers.

Whole Genome Analyses		
Areas	Contributions	Details
Simulation	XSim	drop down origins and positions of chromosomal segments simulate sequence data in descendants in arbitrary pedigrees
Prediction	efficient Gibbs for BayesB	Gibbs sampler without the MH algorithm for the BayesB
	parallel Gibbs for BayesC	parallel sampling of marker effects at each step of the MCMC chain using orthogonal data augmentation
	multiple-trait BayesCπ and BayesB	multiple-trait BayesCπ and BayesB
	single-step Bayesian regression methods	validate single-step Bayesian regression methods using Hanwoo beef cattle data
Validation	efficient LOOCV for Genomic BLUP	efficient LOOCV for Genomic BLUP when $n > p$ or $n < p$

BIBLIOGRAPHY

Aberer, A. J. and Stamatakis, A. (2013). Rapid forward-in-time simulation at the chromosome and genome level. *BMC bioinformatics*, 14(1).

Aguilar, I., Misztal, I., Johnson, D. L., Legarra, A., Tsuruta, S., and Lawlor, T. J. (2010). Hot topic: a unified approach to utilize phenotypic, full pedigree, and genomic information for genetic evaluation of holstein final score. *J Dairy Sci*, 93(2):743–752.

Allen, D. M. (1974). The relationship between variable selection and data agumentation and a method for prediction. *Technometrics*.

Bezanson, J., Edelman, A., Karpinski, S., and Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98.

Browning, S. R. and Browning, B. L. (2007). Rapid and Accurate Haplotype Phasing and Missing-Data Inference for Whole-Genome Association Studies By Use of Localized Haplotype Clustering. *The American Journal of Human Genetics*, 81(5):1084–1097.

Calus, M. P., Schrooten, C., and Veerkamp, R. F. (2014). Genomic prediction of breeding values using previously estimated SNP variances. *Genetics Selection Evolution*, 46(1):52.

Calus, M. P. and Veerkamp, R. F. (2011). Accuracy of multi-trait genomic selection using different methods. *Genetics Selection Evolution*, 43(1):26.

Carlin, B. P. and Chib, S. (1995). Bayesian model choice via Markov-chain Monte-carlo methods. *Journal of the Royal Statistical Society Series B-Methodological*, 57(3):473–484.

Chadeau-Hyam, M., Hoggart, C. J., O'Reilly, P. F., Whittaker, J. C., Iorio, M., and Bald-

ing, D. J. (2008). Fregene: Simulation of realistic sequence-level data in populations and ascertained samples. *BMC Bioinformatics*, 9(1).

Cheng, H., Garrick, D., and Fernando, R. (2015a). XSim: Simulation of Descendants from Ancestors with Sequence Data. *G3 (Bethesda, Md.)*, 5(7):1415–1417.

Cheng, H., Garrick, D. J., and Fernando, R. L. (2016). JWAS: Julia implementation of whole-genome analyses software using univariate and multivariate Bayesian mixed effects model. *avaliable from http://QTL.rocks*.

Cheng, H., Qu, L., Garrick, D. J., and Fernando, R. L. (2015b). A fast and efficient Gibbs sampler for BayesB in whole-genome analyses. *Genetics Selection Evolution*, 47(1):1819.

Crossa, J., de los Campos, G., Pérez, P., Gianola, D., Burgueño, J., Araus, J. L., Makumbi, D., Singh, R. P., Dreisigacker, S., Yan, J., Arief, V., Banziger, M., and Braun, H.-J. (2010). Prediction of Genetic Values of Quantitative Traits in Plant Breeding Using Pedigree and Molecular Markers. *Genetics*, 186(2):713–724.

Daetwyler, H. D., Calus, M. P. L., Pong-Wong, R., de los Campos, G., and Hickey, J. M. (2013). Genomic Prediction in Animals and Plants: Simulation of Data, Validation, Reporting, and Benchmarking. *Genetics*, 193(2):347–365.

de los Campos, G., Gianola, D., and Allison, D. B. (2010). Predicting genetic predisposition in humans: the promise of whole-genome markers. *Nature Reviews Genetics*, 11(12):880–886.

Erbe, M., Hayes, B. J., Matukumalli, L. K., Goswami, S., Bowman, P. J., Reich, C. M., Mason, B. A., and Goddard, M. E. (2012). Improving accuracy of genomic predictions within and between dairy cattle breeds with imputed high-density single nucleotide polymorphism panels. *Journal of Dairy Science*, 95(7):4114–4129.

Fernando, R. (1998). Genetic evaluation and selection using genotypic, phenotypic and pedigree information. In *6th Wld. Cong. Genet. App.Liv. Prod.*, volume 26, pages 329–336, Armidale, Australia.

Fernando, R. and Garrick, D. (2013a). *Genome-Wide Association Studies and Genomic Prediction*, chapter Bayesian Methods Applied to GWAS. Humana Press, New York, NY.

Fernando, R., Toosi, A., Wolc, A., Garrick, D., and Dekkers, J. (2017). Application of Whole-Genome Prediction Methods for Genome-Wide Association Studies: A Bayesian Approach. *Journal of Agricultural, Biological and Environmental Statistics*, 57(4):1–22.

Fernando, R. L., Cheng, H., Golden, B. L., and Garrick, D. J. (2016). Computational strategies for alternative single-step Bayesian regression models with large numbers of genotyped and non-genotyped animals. *Genetics Selection Evolution*, 48(1):96.

Fernando, R. L., Dekkers, J. C., and Garrick, D. J. (2014). A class of Bayesian methods to combine large numbers of genotyped and non-genotyped animals for whole-genome analyses. *Genetics Selection Evolution*, 46(1):50.

Fernando, R. L. and Garrick, D. (2013b). Bayesian Methods Applied to GWAS. In *Genome-Wide Association Studies and Genomic Prediction*, pages 237–274. Humana Press, Totowa, NJ.

Garrick, D., Dekkers, J., and Fernando, R. (2014). The evolution of methodologies for genomic prediction. *Livestock Science*, 166:10–18.

Geyer, C. (1992). Practical Markov chain Monte Carlo. *Stat. Sci.*, 7:473–511.

Ghosh, Joyee and Clyde, Merlise A (2012). Rao–Blackwellization for Bayesian Variable Selection and Model Averaging in Linear and Binary Regression: A Novel Data Augmentation Approach. *Journal of the American Statistical Association*, 106(495):1041–1052.

Gianola, D. (2013). Priors in whole-genome regression: the bayesian alphabet returns. *Genetics*.

Gianola, D., de los Campos, G., Hill, W. G., Manfredi, E., and Fernando, R. (2009a). Additive genetic variability and the bayesian alphabet. *Genetics*, 183(1):347–363.

Gianola, D., de los Campos, G., Hill, W. G., Manfredi, E., and Fernando, R. (2009b). Additive genetic variability and the Bayesian alphabet. *Genetics*, 183(1):347–363.

Gianola, D. and Rosa, G. J. M. (2015). One Hundred Years of Statistical Developments in Animal Breeding. *Annual Review of Animal Biosciences*, 3(1):19–56.

Goddard, M. (2008). Genomic selection: prediction of accuracy and maximisation of long term response. *Genetica*, 136(2):245–257.

Godsill, S. J. (2001). On the relationship between markov chain monte carlo methods for model uncertainty. *Journal of Computational and Graphical Statistics*, 10(2):230–248.

Habier, D., Fernando, R., Kizilkaya, K., and J., G. D. (2010a). Extension of the Bayesian alphabet for genomic selection. In *Proc. 9th World Congress on Genet. Appl. Livest. Prod.*, volume 9, page 468.

Habier, D., Fernando, R. L., and Dekkers, J. C. M. (2007). The Impact of Genetic Relationship Information on Genome-Assisted Breeding Values. *Genetics*, 177(4):2389–2397.

Habier, D., Fernando, R. L., Kizilkaya, K., and Garrick, D. (2011a). Extension of the bayesian alphabet for genomic selection. *BMC Bioinformatics*, 12:186.

Habier, D., Fernando, R. L., Kizilkaya, K., and Garrick, D. J. (2011b). Extension of the bayesian alphabet for genomic selection. *BMC bioinformatics*, 12(1):186.

Habier, D., Tetens, J., Seefried, F.-R., Lichtner, P., and Thaller, G. (2010b). The impact of genetic relationship information on genomic breeding values in german holstein cattle. *Genetics Selection Evolution*, 42(1):5.

Haiminen, N., Utro, F., Lebreton, C., Flament, P., Karaman, Z., and Parida, L. (2013). Efficient in silico chromosomal representation of populations via indexing ancestral genomes. *Algorithms*.

Hao Cheng, D. G. and Fernando, R. (2015). Xsim: Simulation of descendants from ancestors with sequence data. *G3*, 5(7)::1415–1417.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer, New York.

Hayes, B., Bowman, P., Chamberlain, A., Verbyla, K., and Goddard, M. (2009). Accuracy of genomic breeding values in multi-breed dairy cattle populations. *Genetics Selection Evolution*, 41(1):51.

Hayes, B. J., Pryce, J., Chamberlain, A. J., Bowman, P. J., and Goddard, M. E. (2010). Genetic architecture of complex traits and accuracy of genomic prediction: Coat colour, milk-fat percentage, and type in holstein cattle as contrasting model traits. *PLoS Genet*, 6(9):e1001139.

Henderson, C. R. (1984). *Applications of Linear Models in Animal Breeding*. Univ. Guelph, Guelph, Ontario, Canada.

Hoggart, C. J., Chadeau-Hyam, M., Clark, T. G., Lampariello, R., Whittaker, J. C., Iorio, M., and Balding, D. J. (2007). Sequence-Level population simulations over large genomic regions. *Genetics*, 177(3):1725–1731.

Jacob, P., Robert, C. P., and Smith, M. H. (2012). Using Parallel Computation to Improve Independent Metropolis–Hastings Based Estimation. *Journal of Computational and Graphical Statistics*, 20(3):616–635.

Jia, Y. and Jannink, J.-L. (2012). Multiple-Trait Genomic Selection Methods Increase Genetic Value Prediction Accuracy. *Genetics*, 192(4):1513–1522.

Karaman, E., Cheng, H., Firat, M. Z., Garrick, D. J., and Fernando, R. L. (2016). An Upper Bound for Accuracy of Prediction Using GBLUP. *PloS one*, 11(8):e0161054.

Kessner, D. and Novembre, J. (2014). forqs: forward-in-time simulation of recombination, quantitative traits and selection. *Bioinformatics*.

Kizilkaya, K., Fernando, R. L., and Garrick, D. J. (2010). Genomic prediction of simulated multibreed and purebred performance using observed fifty thousand single nucleotide polymorphism genotypes. *J Anim Sci*, 88(2):544–551.

Lee, S. H., Choi, B. H., Lim, D., Gondro, C., Cho, Y. M., Dang, C. G., Sharma, A., Jang, G. W., Lee, K. T., Yoon, D., Lee, H. K., Yeon, S. H., Yang, B. S., Kang, H. S., and Hong,

S. K. (2013). Genome-Wide Association Study Identifies Major Loci for Carcass Weight on BTA14 in Hanwoo (Korean Cattle). *PloS one*, 8(10):e74677.

Legarra, A., Aguilar, I., and Misztal, I. (2009). A relationship matrix including full pedigree and genomic information. *J Dairy Sci*, 92(9):4656–4663.

Lourenco, D. A. L., Misztal, I., Tsuruta, S., Aguilar, I., Ezra, E., Ron, M., Shirak, A., and Weller, J. I. (2014). Methods for genomic evaluation of a relatively small genotyped dairy population and effect of genotyped cow information in multiparity analyses. *Journal of Dairy Science*, 97(3):1742–1752.

Lourenco, D. A. L., Tsuruta, S., Fragomeni, B. O., Masuda, Y., Aguilar, I., Legarra, A., Bertrand, J. K., Amen, T. S., Wang, L., Moser, D. W., and Misztal, I. (2015). Genetic evaluation using single-step genomic best linear unbiased predictor in American Angus. *Journal of animal science*, 93(6):2653–2662.

Maher, B. (2008). The case of the missing heritability. *Nature*, 456:18–21.

Makowsky, R., Pajewski, N. M., Klimentidis, Y. C., Vazquez, A. I., Duarte, C. W., Allison, D. B., and de los Campos, G. (2011). Beyond Missing Heritability: Prediction of Complex Traits. *PLOS Genetics*, 7(4):e1002051.

Manolio, T. A., Collins, F. S., Cox, N. J., Goldstein, D. B., Hindorff, L. A., Hunter, D. J., McCarthy, M. I., Ramos, E. M., Cardon, L. R., Chakravarti, A., Cho, J. H., Guttmacher, A. E., Kong, A., Kruglyak, L., Mardis, E., Rotimi, C. N., Slatkin, M., Valle, D., Whittemore, A. S., Boehnke, M., Clark, A. G., Eichler, E. E., Gibson, G., Haines, J. L., Mackay, T. F., McCarroll, S. A., and Visscher, P. M. (2009). Finding the missing heritability of complex diseases. *Nature*, 461(7265):747–753.

Meuwissen, T. H. E., Hayes, B. J., and Goddard, M. E. (2001a). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157:1819–1829.

Meuwissen, T. H. E., Hayes, B. J., and Goddard, M. E. (2001b). Prediction of Total Genetic Value Using Genome-Wide Dense Marker Maps. *Genetics*, 157(4):1819–1829.

Misztal, I., Aggrey, S. E., and Muir, W. M. (2013). Experiences with a single-step genome evaluation. *Poultry science*, 92(9):2530–2534.

Misztal, I., Tsuruta, S., Strabel, T., and Auvray, B. (2002). BLUPF90 and related programs (BGF90). In *Proceedings of the ....*

Moser, G., Lee, S. H., Hayes, B. J., Goddard, M. E., Wray, N. R., and Visscher, P. M. (2015). Simultaneous Discovery, Estimation and Prediction Analysis of Complex Traits Using a Bayesian Mixture Model. *PLOS Genetics*, 11(4):e1004969.

Nejati-Javaremi, A., Smith, C., and Gibson, J. P. (1997). Effect of total allelic relationship on accuracy of evaluation and response to selection. *J. Anim. Sci.*, 75:1738–1745.

Norris, J. R. (1997). *Markov Chains*. Cambridge series on statistical and probabilistic mathematics. Cambridge University Press, New York.

Park, B., Choi, T., Kim, S., and Oh, S.-H. (2013). National genetic evaluation (system) of hanwoo (korean native cattle). *Asian-Australasian journal of animal sciences*, 26(2):151–156.

Resende, M. F. R., Muñoz, P., Resende, M. D. V., Garrick, D. J., Fernando, R. L., Davis, J. M., Jokela, E. J., Martin, T. A., Peter, G. F., and Kirst, M. (2012). Accuracy of Genomic Selection Methods in a Standard Data Set of Loblolly Pine (Pinus taeda L.). *Genetics*, 190(4):1503–1510.

Saatchi, M., McClure, M., McKay, S., Rolf, M., Kim, J., Decker, J., Taxis, T., Chapple, R., Ramey, H., Northcutt, S., Bauck, S., Woodward, B., Dekkers, J., Fernando, R., Schnabel, R., Garrick, D., and Taylor, J. (2011). Accuracies of genomic breeding values in american angus beef cattle using k-means clustering for cross-validation. *Genetics Selection Evolution*, 43(1):40.

Sahana, G., Guldbrandtsen, B., Janss, L., and Lund, M. S. (2010). Comparison of association mapping methods in a complex pedigreed population. *Genetic Epidemiology*, 34:455–462.

Searle, S. R. (1982). *Matrix Algebra useful for Statistics*. John Wildy and Sons, Inc., New York.

Sorensen, D. A. and Gianola, D. (2002). *Likelihood,Bayesian,and MCMC Methods in Quantitative Genetics*. Springer.

Spiliopoulou, A., Nagy, R., Bermingham, M. L., Huffman, J. E., Hayward, C., Vitart, V., Rudan, I., Campbell, H., Wright, A. F., Wilson, J. F., Pong-Wong, R., Agakov, F., Navarro, P., and Haley, C. S. (2015). Genomic prediction of complex human traits: relatedness, trait architecture and predictive meta-models. *Human Molecular Genetics*, 24(14):4167–4182.

Strandén, I. and Garrick, D. J. (2009). Technical note: Derivation of equivalent computing algorithms for genomic predictions and reliabilities of animal merit. *J Dairy Sci*, 92(6):2971–2975.

Stroustrup, B. (2013). *The C++ Programming Language (4th ed.)*. Addison-Wesley Publishing Company.

Su, G., Christensen, O. F., Janss, L., and Lund, M. S. (2014). Comparison of genomic predictions using genomic relationship matrices built with different weighting factors to account for locus-specific variances. *Journal of dairy science*.

VanRaden, P. M. (2008). Efficient Methods to Compute Genomic Predictions. *Journal of Dairy Science*, 91(11):4414–4423.

VanRaden, P. M., Van Tassell, C. P., Wiggans, G. R., Sonstegard, T. S., Schnabel, R. D., Taylor, J. F., and Schenkel, F. S. (2009). Invited review: reliability of genomic predictions for north american holstein bulls. *J Dairy Sci*, 92(1):16–24.

Vazquez, A. I., de los Campos, G., Klimentidis, Y. C., Rosa, G. J. M., Gianola, D., Yi, N., and Allison, D. B. (2012). A Comprehensive Genetic Approach for Improving Prediction of Skin Cancer Risk in Humans. *Genetics*, 192(4):1493–1502.

Visscher, P. M., Macgregor, S., Benyamin, B., Zhu, G., Gordon, S., Medland, S., Hill, W. G., Hottenga, J.-J., Willemsen, G., Boomsma, D. I., Liu, Y.-Z., Deng, H.-W., Montgomery,

G. W., and Martin, N. G. (2007). Genome Partitioning of Genetic Variation for Height from 11,214 Sibling Pairs. *The American Journal of Human Genetics*, 81(5):1104–1110.

Visscher, P. M., Yang, J., and Goddard, M. E. (2010). A commentary on 'common SNPs explain a large proportion of the heritability for human height' by yang et al. (2010). *Twin Res Hum Genet*, 13(6):517–524.

WANG, H, Misztal, I, Aguilar, I, Legarra, A, and MUIR, W M (2012). Genome-wide association mapping including phenotypes from relatives without genotypes. *Genetics Research*, 94(2):73–83.

Wolc, A., Arango, J., Jankowski, T., Dunn, I., Settar, P., Fulton, J. E., O'Sullivan, N. P., Preisinger, R., Fernando, R. L., Garrick, D. J., and Dekkers, J. C. M. (2014). Genome-wide association study for egg production and quality in layer chickens. *Journal of Animal Breeding and Genetics*, 131(3):173–182.

Wolc, A., Arango, J., Settar, P., Fulton, J., Sullivan, N. P., Preisinger, R., Habier, D., Fernando, R., Garrick, D. J., Hill, W. G., and Dekkers, J. C. M. (2012). Genome-wide association analysis and genetic architcture of egg weight and egg uniformity in layer chickens. *Animal Genetics*, 43 (Suppl 1):87–96.

Wolc, A., Arango, J., Settar, P., Fulton, J. E., O'Sullivan, N. P., Dekkers, J. C. M., Fernando, R., and Garrick, D. J. (2016). Mixture models detect large effect QTL better than GBLUP and result in more accurate and persistent predictions. *Journal of Animal Science and Biotechnology*, 7(1):7.

Wu, X.-L., Sun, C., Beissinger, T. M., Rosa, G. J., Weigel, K. A., Gatti, N. d., and Gianola, D. (2012). Parallel Markov chain Monte Carlo - bridging the gap to high-performance Bayesian computation in animal breeding and genetics. *Genetics Selection Evolution*, 44(1):29.